

ZÁPADOČESKÁ UNIVERZITA V PLZNI

FAKULTA EKONOMICKÁ

Bakalářská práce

**Automatizace postupů při zpracování dat z účetního
deníku pro účely auditu**

**Bookkeeping journal's data processing automation for
financial audit**

Jakub Stelzer

Plzeň 2015

VLOŽIT ZADÁNÍ

Prohlašuji, že jsem bakalářskou práci na téma

„Automatizace postupů při zpracování dat z účetního deníku pro účely auditu“

vypracoval samostatně a pod odborným dohledem vedoucího bakalářské práce za použití pramenů uvedených v příložené bibliografii.

Plzeň dne

.....

podpis autora

Poděkování:

Rád bych poděkoval panu Ing. Mgr. Milanu Svobodovi, Ph.D. za odborné vedení mé bakalářské práce, cenné rady a připomínky, které mi poskytoval v průběhu tvorby práce, a které mi pomohly k jejímu úspěšnému dokončení. Jeho pomoci si velmi vážím. Dále bych chtěl poděkovat HZ Plzeň spol. s r. o., která mi poskytla potřebná data pro tvorbu této práce.

Obsah

Obsah	5
Úvod.....	7
1. Základní terminologie.....	8
1.1. Pojmy a informační technologie	10
1.1.1. Technologie	10
1.1.2. Informační systémy.....	11
1.1.3. ERP	11
1.1.4. Modul.....	11
1.1.5. CRM.....	12
1.1.6. Účetní software	12
1.1.7. Vnitropodnikové (podnikové) procesy	12
1.1.8. Java (programovací jazyk).....	13
1.1.9. Visual Basic (VB).....	13
1.1.10. Microsoft Excel.....	13
1.2. Charakteristika problematiky	14
1.3. Roztříštěnost informačních systémů	14
1.3.1. Vývoj přírůstku českého trhu.....	15
1.3.2. Tržní podíly all-in-one ERP systémů podle počtu implementací	16
1.4. Způsob zpracování dat auditorů	19
2. Automatizace postupů při zpracování dat.....	20
2.1. Rozbor vstupních dat.....	20
2.1.1. SAP	20
2.1.2. K2.....	21
2.1.3. JUMP	22
2.1.4. Vhodný výstup po automatizaci zpracování dat	22

2.2.	Možnosti řešení automatizace	23
2.2.1.	Visual Basic for Applications	23
2.2.2.	JAVA	25
2.2.3.	Modul do ERP	28
2.2.4.	Shrnutí možností automatizace	28
2.3.	Návrh a dokumentace skriptu.....	29
2.3.1.	Systém SAP	29
2.3.2.	Systém JUMP	41
2.3.3.	Postup zpracování	51
2.3.4.	Výsledné filtrování	52
3.	Zhodnocení efektivnosti navrženého řešení	54
	Závěr	55
	Seznam obrázků	56
	Seznam tabulek	57
	Seznam grafů	58
	Seznam použité literatury	59
	Publikace:.....	59
	Elektronické zdroje:	60
	Abstrakt.....	61
	Abstract.....	62

Úvod

Bakalářská práce se zabývá tématem automatizace postupů při zpracování dat z účetních deníků pro účely auditu. Téma jsem zvolil proto, že jsem se chtěl zabývat problematikou, do které mohu vnést své vlastní myšlenky a postup, a při které se mohu mnohému naučit. Problematika zpracování dat a její automatizace pro mne byla velmi lákavá. Jistou zkušenost s programováním a prací s daty v programovacích jazycích Java, PHP a C jsem již měl a možnost přidat nebo rozšířit znalost programovacího jazyka Visual Basic a pokročilejší práce s daty v Excelu mne dále motivovala k výběru tohoto tématu.

Společností, která poskytla výstupní data z účetních deníků, a tím umožnila tuto práci uskutečnit a definovat její téma, je HZ Plzeň spol. s r. o. Tato společnost je dceřinou společností auditorské společnosti HZ Praha, spol. s r. o., jež je členem mezinárodní sítě ekonomicko-poradenských společností PKF se sídlem v Londýně, která byla založena v roce 1969. Společnost HZ Plzeň, spol. s r. o. poskytuje služby v oblasti účetního a daňového auditu, účetního a ekonomického poradenství. Klienty společnosti jsou podnikatelé v oblasti výroby, obchodu i služeb, územně samosprávné celky, příspěvkové organizace a nadace.

Tato práce je rozdělena do dvou částí, na část teoretickou a část praktickou. Teoretická část se zabývá definováním základní terminologie, úvodem do informačních systémů a charakterizováním problematiky zpracování dat. Dále jsou v této části probírány informační systémy, jejich rozmanitost a rozdělení mezi podniky podle využití.

Praktická část je zaměřena na samotné řešení automatizace postupů při zpracování dat. Nejprve jsou rozebrána vybraná vstupní data, která poskytla HZ Plzeň spol. s r. o. V další části jsou rozebírány jednotlivé možnosti zpracování dat, nástroje a programovací jazyky, ve kterých je možné danou problematiku řešit. Stěžejní částí práce jsou návrh, realizace a dokumentace navrženého řešení automatizace postupů při zpracování dat z účetních deníků pro účely auditu. V posledním oddíle jsou shrnuty dosažené výsledky, výstupní data, jejich srozumitelnost, efektivnost zvoleného řešení, možná zlepšení a modifikace realizace zvoleného řešení.

1. Základní terminologie

Nacházíme se v 21. století a všude kolem nás jsou moderní technologie, které člověk vynalezl a využívá, aby sobě a svému okolí usnadnil práci. Tyto technologie jsou založeny na matematických či fyzikálních principech, které se ve světě vyskytují. Moderní technologie, zejména ty informační, usnadňují lidem práci na základě výpočtů, které stroj zvládá daleko lépe, než průměrný člověk. Informační technologie a informační systémy nahrazují potřebu lidského snažení v různých odvětvích, ať už ve zdravotnictví, vojenství nebo obchodu. Žijeme v době, kdy je využití těchto technologií tak běžné, že nefunkčnost, porucha či špatné využití může mít za následek kritické selhání celého záměru podpořeného tímto systémem. Jak si koupit vodu z nápojového automatu s vadným či nefunkčním kódem? Jak se dostat z placeného parkoviště, když vás při výjezdu systém nerozezná a brána se neotevře? To jsou otázky, se kterými se budeme muset v blízké budoucnosti vyrovnat. Na tyto konkrétní otázky se ovšem nabalují další, na které budu v této práci hledat odpovědi.

Správné využívání informačních technologií a technologií obecně může ve značné míře usnadnit práci. Už první počítače byly zkonstruovány proto, aby usnadnily výpočty. Využití počítačů a informačních technologií může ušetřit hodiny, dny a roky času, které mohou být věnovány důležitějším činnostem.

Informační technologie a informační systémy pracují s informacemi. Informace je jedním z hlavních zdrojů v podnikání. Způsobů definování pojmu informace je v literatuře k nalezení celá řada. Tvůrci těchto definic se rozcházejí hlavně v různých úrovních pohledu, jimiž lze na informaci nahlížet. Syntaktický pohled na informaci je orientován zejména na vnitřní strukturu informace s tím, že sleduje znaky, které informace nese. Takový pohled nesleduje do takové míry vztah informace k jejímu příjemci. Sémantický pohled sleduje zejména obsah samotné informace a také do jisté míry přehlíží vztah informace k jejímu příjemci. Pragmatický pohled se od přechozích dvou uvedených v mnohém rozchází. Takový pohled na informace je zaměřen na praktické využití informace a tím i na význam pro jejího příjemce. Pragmatický pohled nahlíží na informaci jako na nezbytnou součást rozhodovacího procesu, ať už v podnikání nebo ve společenském a osobním životě.¹

¹ SODOMKA, Petr a Hana KLČOVÁ. Informační systémy v podnikové praxi. 2. aktualiz. a rozš. vyd. Brno: Computer Press, 2010, 501 s. ISBN 978-80-251-2878-7.

Peter Drucker ve své knize Postkapitalistická společnost² z roku 1993 tvrdí, že samotná informace je tím jediným smysluplným zdrojem pro podnikání. Podle Druckera se ostatní výrobní faktory stávají druhořadými. Jeho slavný výrok nese podobné sdělení: „Když něco nemůžete změřit, nemůžete to ani řídit.“. Tento výrok se týkal jeho moderního přístupu k managementu, ale každé manažerské rozhodnutí je vykonáno na základě informací.

V uplynulých letech bylo v podnicích znatelné zavádění komplexních informačních systémů z kategorie Enterprise Resource Planning(ERP)³. Aplikace ERP lze považovat za ty, které nejvýrazněji ovlivňují současný svět podnikání. Využívá je totiž více než 90% podniků zařazených v Čechách v TOP100⁴. Při pohledu na náročnost na lidské zdroje, finanční a časovou náročnost implementací a provozování ERP je zřejmé, že podnikové informační systémy mají ekonomický i společenský význam.

Informační systémy podporují a propojují jednotlivé vnitropodnikové složky. Dnešní podnikové informační systém neřeší pouze úlohy spojené s důležitými podnikovými funkcemi, jakými jsou například finance, personalistika, plánování, prodej, nákup, logistika včetně e-businessu a e-commerce. Informační systém musí držet krok s potřebami podnikání. Takovými potřebami mohou být různé fáze, trvalé požadavky na podporu efektivnosti, flexibility a inovace hlavních podnikových procesů, produktů a služeb. Správná implementace a užívání podnikového informačního systému je stěžejní pro jeho efektivnost.

Během posledních desítek let vznikaly různé informační systémy. Některé byly vytvářeny na míru určitým společnostem a jiné byly vytvářeny jako univerzální systém za účelem prodeje a nasazení do více podniků. Každý informační systém byl navrhován jinak a každý se svou podobou, funkcionalitou i procesy liší. Odlišnost v začátcích se promítla i do úplného závěru, za který mohou být považována výstupní data z informačního systému a jejich napojení na další části informačního systému. Výstupní data se liší ve značné míře, protože každý systém pracuje s daty rozdílným způsobem. To se týká i účetních softwarů nebo účetních modulů. „*Účetní jednotky jsou povinny vést*

² DRUCKER, Peter F. D. Postkapitalistická společnost. 1.vyd. Praha: Management Press, 1993, 197 s. ISBN 80-856-0331-4.

³ Definováno v části s pojmy.

⁴ BASL, Josef a Roman BLAŽÍČEK. Podnikové informační systémy: podnik v informační společnosti. 3., aktualiz. a dopl. vyd. Praha: Grada, 2012, 323 s. Management v informační společnosti. ISBN 978-80-247-4307-3.

*účetnictví tak, aby účetní závěrka sestavená na jeho základě podávala věrný a poctivý obraz předmětu účetnictví a finanční situace účetní jednotky.*⁵ Účetnictví je v mnohých případech potřeba dále studovat a rozebírat. Jedním z případů může být finanční audit, který má za úkol nezávislé ověření účetních výkazů. Výsledkem ověření je vyslovení názoru, zda jsou předložené účetní výkazy pravdivé a věrné a zda jsou v souladu s odpovídajícími předpisy.

Při účetním auditu je nahlíženo do dokumentů a účetních výkazů, které jsou zmíněnými výstupy účetních softwarů. Tato práce si dává za cíl zjednodušit tyto výstupy a převést je do takové podoby, aby byl účetní audit zjednodušen.

1.1. Pojmy a informační technologie

Pro porozumění stěžejní části práce, kterou je návrh a realizace vybraného řešení automatizace, a možnost běžného využití terminologie, týkající se problematiky zpracování dat a informačních systému, je nutné vymezit základní pojmy. Zmíněna jsou jak základní a běžné pojmy, tak konkrétnější a specializovanější pojmy, které se využívají v daném kontextu.

1.1.1. Technologie

Využití technologií je v dnešním světě tak běžné, že samotný význam slova už nemusí být tak známý. Ať už se jedná o výrobky vyrobené určitou technologií nebo o námi využívané technologie. Technologií je tramvaj, kterou se dopravují tisíce studentů do škol a lidí za prací. Technologií je mobilní telefon, bez kterého si dnes jen málokdo dovede představit normální den. Technologií je i výrobní proces na zpracování oblečení, které má na sobě každý.

Za technologii je považováno odvětví techniky, které se zabývá tvorbou, zaváděním a zdokonalováním výrobních postupů. Technologií tedy není pouze finální produkt. I tento finální produkt byl vyroben jistou technologií, která byla vynalezena nebo alespoň využita pro ten jistý účel.

⁵ Zákon o účetnictví (zákon č. 563/1991 Sb.) § 7

1.1.2. Informační systémy

V této práci se často píše o informačních systémech. Tyto systémy jsou využívány v určitých společnostech nebo v určitých oblastech lidského života natolik, že je snadné je přehlížet.

Informační systém (IS) je soubor jednotlivců, lidí, technologických prostředků, způsobů a metod, které zajišťují sběr, přenos, transformaci, zpracování a uchování dat za účelem pozdější prezentace a předání dat do rukou a podle potřeb uživatele.⁶ Příkladem informačního systému může být kartotéka, telefonní seznam, správa došlé pošty nebo účetnictví. Systém nemusí být nutně automatizovaný pomocí počítačů a může být i v papírové podobě. Podpoření informačního systému výpočetní technikou usnadňuje práci s daty a využívá veškerých výhod, které počítače nabízejí.

1.1.3. ERP

ERP neboli Enterprise Resource Planning integrují podnikové procesy napříč celým podnikem účetní jednotky nebo společnosti.⁷ ERP II (nebo ERP 2) rozšiřují dodavatelskou funkcionalitu vůči externím společnostem pro snížení nákladů, rozšiřují dodavatelské řetězce a zefektivňují je a jdou naproti vzájemným inovacím.

„ERP systémy představují softwarové nástroje používané k řízení podnikových dat. ERP systémy pomáhají podnikům v oblasti dodavatelského řetězce, příjmu materiálu, skladového hospodářství, přijímání objednávek od zákazníků, plánování výroby, expedice zboží, účetnictví, řízení lidských zdrojů a v dalších podnikových funkcích. (Somers and Nelson, 2003)“⁸

1.1.4. Modul

Modul, resp. funkční oblast, je programovou částí informačního systému. Informační systém se skládá z modulů. Tyto moduly dohromady tvoří celek informačního systému. Mezi jednotlivé moduly mohou například patřit modul účetnictví, modul zásob, modul CRM, modul Business intelligence.

⁶ SODOMKA, Petr a Hana KLČOVÁ. Informační systémy v podnikové praxi. 2. aktualiz. a rozš. vyd. Brno: Computer Press, 2010, 501 s. ISBN 978-80-251-2878-7.

⁷ WOOD, Bill. ERP vs. ERP II vs. ERP III Future Enterprise Applications. [online]. [cit. 2015-03-15]. Dostupné z: <http://www.r3now.com/erp-vs-erp-ii-vs-erp-iii-future-enterprise-applications/>

⁸ BASL, Josef a Roman BLAŽÍČEK. Podnikové informační systémy: podnik v informační společnosti. 3., aktualiz. a dopl. vyd. Praha: Grada, 2012, 323 s. Management v informační společnosti. ISBN 978-80-247-4307-3.

Jednotlivé moduly, které daný informační systém ve společnosti obsahuje a využívá, jsou často vybírány a konfigurovány společností na míru. To znamená, že různé společnosti využívají různé moduly informačních systémů. Pokud má firma působení hlavně ve službách, pak pravděpodobně nebude muset využívat tak obsáhlý modul skladů a zásob jako například společnost, která se zabývá zejména prodejem a nákupem materiálu a zboží. Základní moduly se ovšem opakují téměř v každém informačním systému nebo ekonomickém softwaru. Mezi základní moduly patří zejména modul účetnictví.

1.1.5. CRM

CRM je zkratkou pro Customer Relationship Management nebo česky řízení vztahů se zákazníky. Jedná se o zákaznický orientovaný management, o podnikatelský přístup, který se vyznačuje aktivní tvorbou a udržováním dlouhodobě prospěšných vztahů se zákazníkem. Právě orientace na potřeby a ziskovost zákazníků generuje poptávku po automatizaci externích procesů (obchodu, marketingu, servisních služeb a řízení kontaktů), a tedy po uplatnění CRM systémů.⁹ Aby se vztah mezi společností a zákazníkem mohl považovat za CRM, musí být prospěšný pro obě strany. To vylučuje neetické chování k zákazníkům. „*Převážná většina autorů píšících o CRM zdůrazňuje především nutnost uspokojovat potřeby zákazníků.*“¹⁰

1.1.6. Účetní software

Účetní software je takový software¹¹, který představuje sadu nástrojů a databází pro vedení účetnictví v digitální podobě. Tento účetní software je rozšířen mezi jednotlivce, obchodní společnosti, příspěvkové organizace i nestátní neziskové organizace. Většinou se jedná o modulární aplikaci, kterou lze přizpůsobit předmětu podnikání a specifickým požadavkům uživatele.

1.1.7. Vnitropodnikové (podnikové) procesy

Podnikový proces (obchodní proces nebo jen zkráceně proces) je jistý tok práce nebo činností. Proces se běžně definuje jako "*transformace vstupů do konečného produktu/služby prostřednictvím aktivit přidávajících tomuto produktu hodnotu*". Hammer, M., Champy, J., 1996 definují proces takto: „*Proces je soubor činností, který*

⁹ SODOMKA, Petr a Hana KLČOVÁ. Informační systémy v podnikové praxi. 2. aktualiz. a rozš. vyd. Brno: Computer Press, 2010, 501 s. ISBN 978-80-251-2878-7.

¹⁰ Tamtéž.

¹¹ Softwarem se rozumí počítačový program používaný v počítači, který provádí určitou činnost.

vyžaduje jeden nebo více druhů vstupů a tvoří výstup, který má pro zákazníka hodnotu“. Procesně řízený podnik je tedy orientován na výsledek všech činností, tj. na hodnotu, za kterou je zákazník ochoten zaplatit. Norma ISO 9001 definuje proces takto: „*Soubor vzájemně působících činností, který přeměňuje vstupy na výstupy.*“

1.1.8. Java (programovací jazyk)

Java je programovací jazyk a výpočetní platforma poprvé vydaná společností Sun Microsystems v roce 1995. Existuje spousta aplikací a webových stránek, které by nefungovaly, kdyby nebyla nainstalována Java. Každý den jsou nové webové stránky a aplikace s potřebou Javy vytvářeny. Java je rychlý, bezpečný a spolehlivý programovací jazyk. Java je využívána v notebookech i v datových centrech, v herních konzolách i ve vědeckých počítačích, v mobilních telefonech i všude na internetu.¹²

1.1.9. Visual Basic (VB)

Visual Basic je vytvořen pro produktivní vytváření typově bezpečných a objektově orientovaných aplikací. Visual Basic umožňuje vývojářům vyvíjet pro operační systém Windows, web nebo mobilní zařízení. Stejně jako všechny jazyky zaměřené na Microsoft .NET Framework, programy napsané v jazyce Visual Basic profitují ze zabezpečení a ze vzájemné funkční spolupráce.¹³

1.1.10. Microsoft Excel

Microsoft Excel je program, který obsahuje nástroje na vytvoření tabulky a výpočet a analýzu údajů. V programu MS Excel je možné vytvářet tabulky, které automaticky vypočítají součty zadaných číselných údajů, může vytvořit tabulky s elegantním rozložením a vytvořit jednoduché grafy. Program MS Excel je součástí balíku MS Office. Tento balík produktů obsahuje různé typy softwaru určené na vytváření dokumentů, pracovních archů, prezentací a správu emailů.¹⁴

V počátcích vývoje musel být Excel označován jako „Microsoft Excel“ v důsledku sporů o ochrannou známku, později se běžně přešlo k používání prostého označení „Excel“.

¹² What is Java technology and why do I need it?. Java.com. [online]. [2015] [cit. 2015-03-21]. Dostupné z: https://www.java.com/en/download/faq/whatis_java.xml

¹³ Visual Basic. Microsoft Development Network. [online]. [2013] [cit. 2015-03-22]. Dostupné z: <https://msdn.microsoft.com/cs-cz/library/2x7h1hfk.aspx>

¹⁴ Co je program Excel?. Support Office. [online]. [2015] [cit. 2015-03-25]. Dostupné z: <https://support.office.com/sk-sk/article/%C4%8Co-je-program-Excel-8373c3d7-bd64-4b7f-bdbd-1fa4b2007b09?ui=sk-SK&rs=sk-SK&ad=SK>

1.2. Charakteristika problematiky

Společnost HZ Plzeň spol. s r. o., která poskytuje služby v oblasti účetního auditu, daňového, účetního a ekonomického poradenství, požádala o pomoc s automatizací zpracováním dat. Jejich auditoři se zabývají analýzou účetních výkazů, ve kterých na základě svých pravidel vyhledávají jisté znaky. Aby analýzu mohli provádět, potřebují výstupní data z účetních či ekonomických softwarů nebo informačních systémů svých klientů. Klienti společnosti využívají nejrůznější systémy. Těmito systémy jsou například ABRA, CSS, ESO, HB, Helios, IMIS, JUMP, K2, Money S3, MRP, Pohoda, Quit, SAP nebo Stereo. Každý z uvedených informačních systémů a jejich výstupy z účetních deníků zobrazují data jiným způsobem (viz kapitola Automatizace postupů při zpracování dat). HZ Plzeň spol. s r. o. poskytla ke zpracování výstupní data z účetních softwarů několika nejmenovaných společností.

Základní problém, se kterým se HZ Plzeň spol. s r. o. potýkala, byl v několikařádkových zápisech jedné účetní operace. Takové zápisy komplikují čtení a samotnou analýzu dat. Pod víceřádkovým zápisem není skryto pouze rozepsání jedné účetní operace příslušným účtům hlavní knihy, ale také rozepsání jedné operace do stejného sloupce. Tato problematika je dále zpracována v části s Návrhem a dokumentací skriptu.

Aby bylo možné data zpracovat, je potřeba navrhnout systém automatizace, kterým může být například vytvoření jistého programu či skriptu, který bude data do vhodnější podoby upravovat. Vytvořením návrhu automatizace se zabývá kapitola Automatizace postupů při zpracování dat.

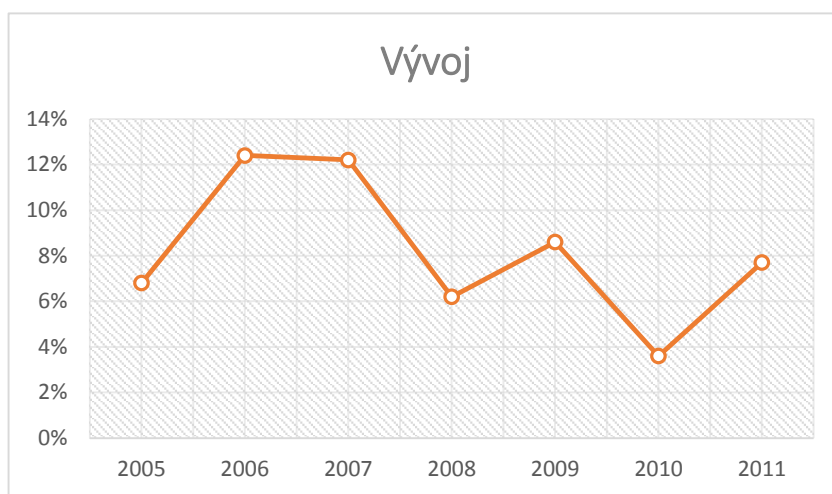
1.3. Roztříštěnost informačních systémů

Trh s informačními systémy je trhem, kde se každý účastník snaží prodat svůj nejlepší produkt za pro něj nejlepší cenu za účelem největšího zisku. Díky tomu se na trhu s informačními technologiemi vyskytuje nespočet různých informačních systémů, které mezi sebou často soupeří, což se projevuje na absolutní vzájemné nekompatibilitě. Jsou k nalezení i systémy, které navzájem spolupracují a snaží se propojovat výstupy a vstupy na obou stranách. Takové snažení se projevuje jistou kompatibilitou, která ovšem nemusí být vždy úplná. Stejným způsobem trpí data, která tyto informační systémy využívají a která jsou jejich výstupem. Každý informační systém má svůj vlastní výstup, který může sloužit jako vstup do dalšího modulu v rámci celého informačního systému nebo jako

hotová data, která se ukládají, zpracovávají, čtou nebo analyzují. Ve výsledku jsou tyto systémy natolik rozdílné, že společnosti, které se zabývají zpracováním těchto výstupních dat, mají problémy s jejich zpracováním.

1.3.1. Vývoj přírůstku českého trhu

V posledních letech došlo k výraznému nárůstu využívání ERP systémů v podnicích. V grafu číslo 1 jsou zachyceny každoroční relativní přírůstky nových ERP projektů ve všech segmentech trhu. Centrum pro výzkum informačních systémů (CVIS) provádí hodnocení českého ERP trhu již od roku 2000.¹⁵



Graf 1 - Vývoj přírůstku českého ERP trhu v letech 2005 až 2011. Graf znázorňuje každoroční relativní přírůstek nových ERP projektů (referencí) ve všech segmentech trhu. Zdroj: CVIS

„Spolehlivost výzkumu je založena na kombinaci kvantitativní s kvalitativní formy šetření. Ta spočívá v průběžné realizaci projektivních rozhovorů a případových studií, jichž jsme do konce roku 2011 realizovali více jak 120.“¹⁶ Kvalitativní výsledky jsou dále tříděny podle jednotlivých odvětví, velikostí firem, použitých IT aplikací a dalších parametrů. CVIS zajišťuje spolehlivost výsledků také porovnáním aktuálních údajů s těmi z přechozích let a daty z externích zdrojů, při němž jsou odhaleny extrémní odchylky nebo chyby respondentů. Z grafu je patrné, že každoročně relativně přibývá zhruba 8% nových ERP projektů.

¹⁵ SODOMKA, Petr a Hana KLČOVÁ. Český trh ERP zrychlil růst: V malých a středních podnicích přibýlo dva tisíce projektů. [online]. [cit. 2015-02-24]. Dostupné z: <http://www.systemonline.cz/erp/cesky-trh-erp-zrychlil-rust.htm>

¹⁶ Tamtéž.

1.3.2. Tržní podíly all-in-one ERP systémů podle počtu implementací

All-in-one ERP systémy jsou souhrnným označením pro takové ERP systémy, které nabízejí podporu celého podnikání informačním systémem. Mezi takové systémy tedy nepatří drobné účetní softwary a samostatné moduly informačních systémů.

Příkladem takového all-in-one ERP systému může být například SAP. SAP na rozdíl od dobře známých desktopových aplikací, mezi něž patří například MS Word či Excel, používaných jednotlivci k provádění úkolů, je souborem podnikových aplikací. Ty jsou používány jednotlivci k řízení finančního účetnictví celých firem, ke správě celých skladů a distribučních středisek, k vyhledávání způsobů rychlejšího prodeje produktů, ke zpracování mezd ve firmě apod.¹⁷

Český statistický úřad vydává průzkum s názvem Informační společnost v číslech. Data, která průzkum obsahuje, vycházejí z výsledků Ročního statistického šetření o využívání ICT v podnikatelském sektoru (ICT 5-01), které provádí Český statistický úřad již od roku 2002. V tomto průzkum se nachází i data o podnicích používající vybrané IS.

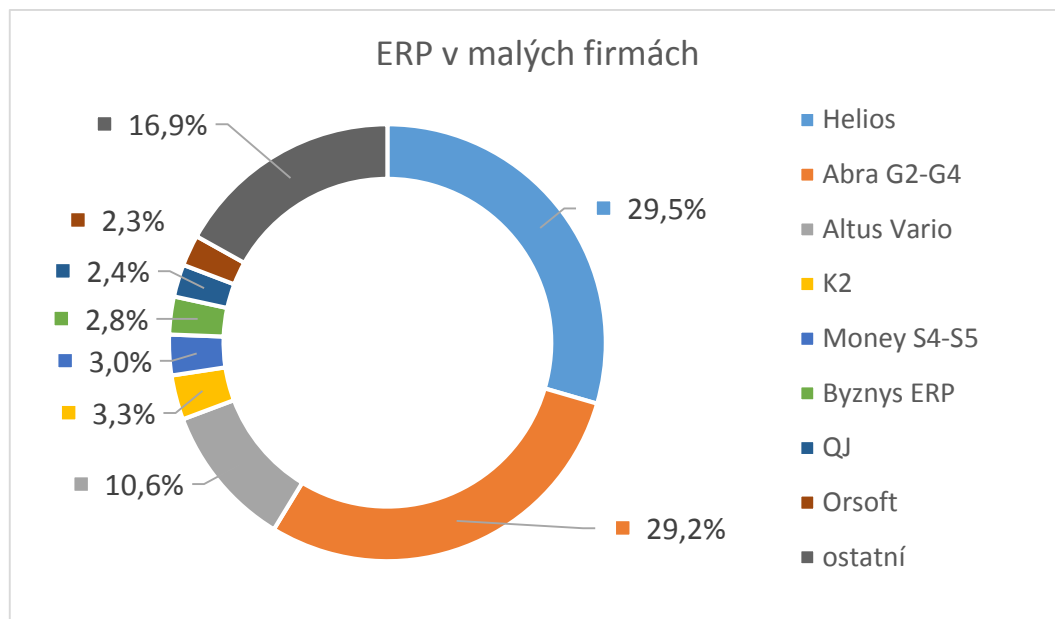
	ERP	CRM	RFID ¹⁸
Celkem (10 a více zaměstnanců)	27,7	19,1	5,9
Malé (10-49 zaměstnanců)	19,5	14,7	3,3
Střední (50-249 zaměstnanců)	53,3	34,3	13,4
Velké (250 a více zaměstnanců)	81,1	41,9	26,6

Tabulka 1 - Podniky používající vybrané IS; leden 2014 (hodnoty v %). Zdroj: <http://www.czso.cz>

Z tabulky 1 je patrné, že větší společnosti využívají informační systémy ERP ve značně větší míře než malé. Výběr a implementace různých informačních systémů závisí na velikosti podniku. Menší podniky využívají menší a jednodušší informační systémy. Velké podniky často využívají komplexních all-in-one ERP systémů. Tento stav je zobrazen v následujících grafech.

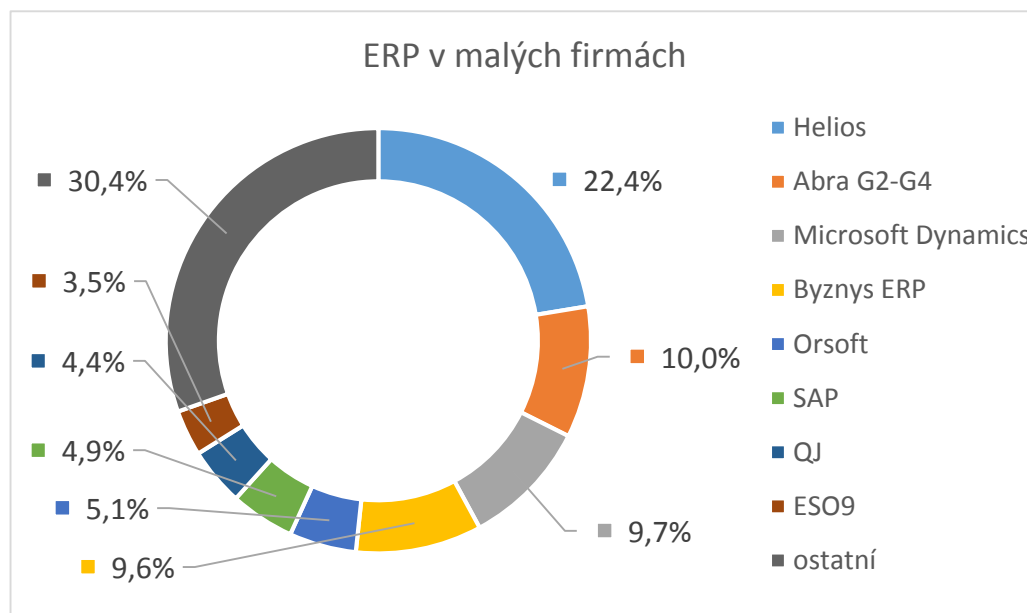
¹⁷ ANDERSON, George W. Naučte se SAP za 24 hodin. 1. vyd. Brno: Computer Press, 2012, 432 s. ISBN 978-80-251-3685-0.

¹⁸ Identifikátor navržený k identifikaci zboží navazující na systém čárových kódů. Slouží k bezkontaktní komunikaci na krátkou vzdálenost.



Graf 2 - Hodnoceno 62 all-in-one ERP systémů nasazených v malých organizacích v ČR (od 10 do 49 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 9 829 referencí. (Zdroj: CVIS 2012)

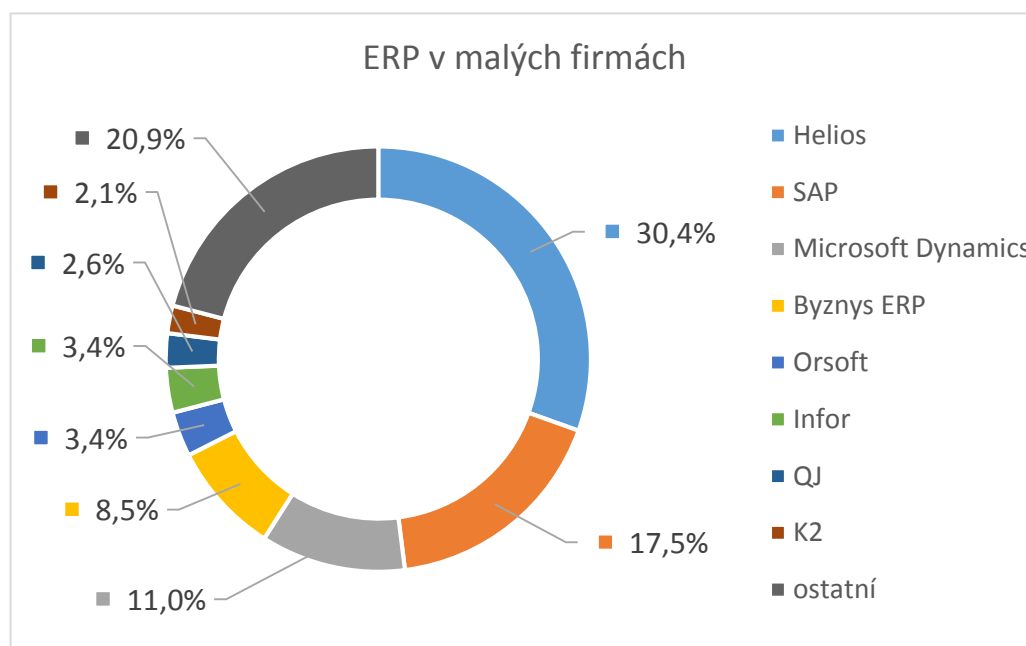
Na grafu číslo 2 jsou vyobrazeny all-in-one ERP systémy implementované v malých organizacích v České republice. Šetření bylo provedeno celkem v 9 829 společnostech nebo podnicích. V malých organizacích je nejvíce využity Helios a ABRA.



Graf 3 - Hodnoceno 62 all-in-one ERP systémů nasazených ve středně velkých organizacích v ČR (od 50 do 249 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 7 324 referencí. (Zdroj: CVIS 2012)

Na grafu číslo 3 jsou vyobrazeny all-in-one ERP systémy implementované ve středně velkých organizacích v České republice. Šetření bylo provedeno celkem v 7 324 společnostech nebo podnicích. Středně velké organizace využívají nejvíce systém

Helios a systém ABRA. Organizace využívají i Microsoft Dynamics, který je často k vidění ve státním sektoru. Začíná se zde objevovat i systém SAP.



Graf 4 - Hodnoceno 62 all-in-one ERP systémů nasazených ve velkých organizacích v ČR (od 250 do 1 000 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 3 281 referencí. (Zdroj: CVIS 2012)

Na grafu číslo 4 jsou vyobrazeny all-in-one ERP systémy implementované ve velkých organizacích v České republice. Šetření bylo provedeno celkem v 3 281 společnostech nebo podnicích. U těchto větších firem je již patrný větší výskyt informačního systému SAP. Ve všech uvedených velikostech podniků v ČR je nejčastěji využíván informační systém Helios.

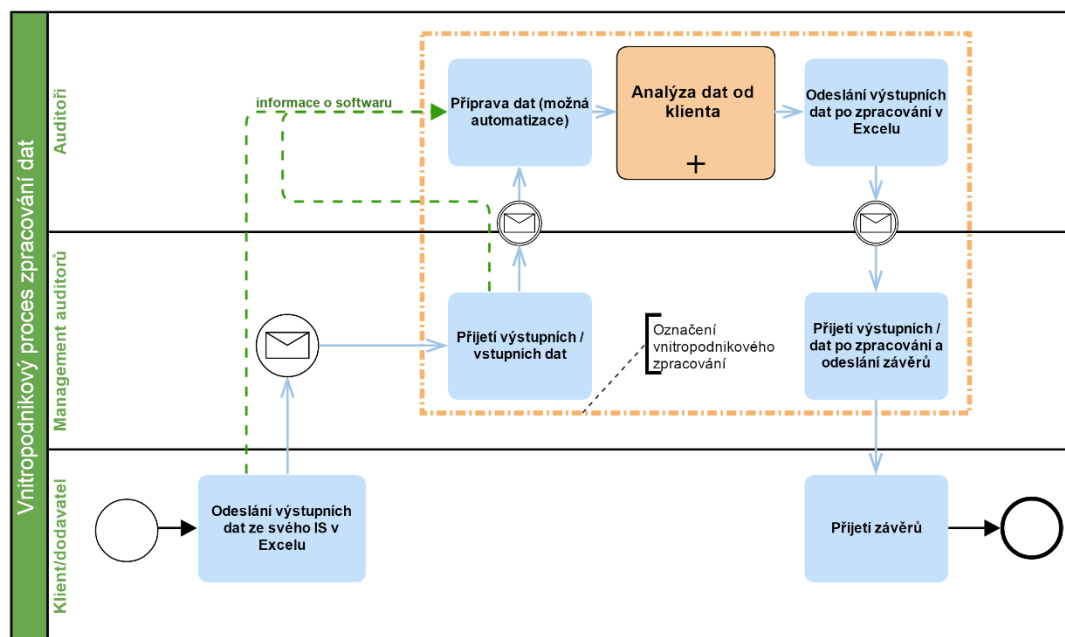
Na zrychlení růstu českého ERP trhu se v posledních letech nejvíce podílely malé a střední podniky. Podniky s počtem zaměstnanců menším než deset dávají často přednost modernímu ERP řešení před ekonomickým informačním systémem. V mikrofirmách jsou úspěšné zejména produkty Abra G2, G3, Money S4 a částečně i Money S5. Abra eviduje více jak tisíc mikrofírem¹⁹ mezi svými zákazníky užívajícími plnohodnotné ERP produkty, u Cígler Software, výrobce Money S4 a S5, jsou jich desítky.²⁰

¹⁹ Společnost s počtem zaměstnanců menší než deset.

²⁰ SODOMKA, Petr a Hana KLČOVÁ. Český trh ERP zrychlil růst: V malých a středních podnicích přibýlo dva tisíce projektů. [online]. [cit. 2015-02-24]. Dostupné z: <http://www.systemonline.cz/erp/cesky-trh-erp-zrychlil-rust.htm>

1.4. Způsob zpracování dat auditorů

Aby bylo jasné, jakým způsobem data v auditorské společnosti postupují, je potřeba tento proces vysvětlit. Každý klient auditorské společnosti využívá jiný účetní software a pracuje rozdílným způsobem. Auditori nahlíží do jednotlivých výstupů z účetních modulů informačních systémů nebo účetních softwarů.



Obrázek 1 - Diagram popisující způsob zpracování dat auditorů

Na obrázku číslo 1 je vyobrazen diagram popisující způsob, jakým postupují výstupní data účetních softwarů podnikem, který se zabývá zpracováním nebo čtením těchto dat. Klient své výstupy z účetního softwaru buď odesílá sám nebo s pomocí informatiků. Tyto data z IS jsou uloženy jako dokument v MS Excelu. V některých případech mohou být i v MS Accessu, ale ty se dají do Excelu snadno převést. V následujícím kroku odesílá klient data ze svého informačního systému auditorům, kteří tato data přijímají. Během těchto kroků by samotný auditor nebo osoba zodpovědná za zpracování dat měla dostat informace, o jaký software se jedná. Tyto výstupní data z účetního softwaru se pro auditory stávají vstupními daty vlastní analýzy. Aby tato analýza mohla probíhat rychleji, je potřeba data jistým způsobem připravit. V tomto kroku je možnost využití jistě automatizace. Po úspěšném převedení mohou být data rychleji a snadněji analyzována a odeslána zpět managementu auditorů. Zmíněnou automatizací se zabývá následující kapitola Automatizace postupů při zpracování dat.

2. Automatizace postupů při zpracování dat

Aby bylo možné vytvořit funkční program nebo skript, který bude automatizovat zpracovávání dat, je potřeba analyzovat vstupní data a posoudit možnosti řešení automatizace. Následně může být na základě závěrů z analýz vytvořen ve vybraném softwaru či programovacím jazyce návrh a funkční řešení. U řešení bude následně provedeno zhodnocení efektivnosti a posouzení, zda je takové řešení vhodné a funkční.

2.1. Rozbor vstupních dat

Veškerá vstupní data, která byla firmou HZ Plzeň spol. s r. o. poskytnuta, jsou soubory z aplikace MS Excel. Výběr z nich bude sloužit jako vzorek pro analýzu vstupních dat do programu či skriptu. Pro vysvětlení je potřeba uvést, že v práci jsou uváděna místa výstupní a místa vstupní data. Výstupními daty se rozumí výstupní data z účetních softwarů a ekonomických informačních systémů a vstupními daty se rozumí vstupní data do skriptu, který tyto data zpracovává. Ve své podstatě se jedná o tytéž data, jejich pojmenování se liší pouze podle úrovně pohledu.

Jednotlivá vstupní data jsou v následující analýze rozebrána pouze obecně. Jejich podrobná analýza a vysvětlení významu jednotlivých položek jsou prováděny až v návrhu jejich zpracování. V teoretické části bylo uvedeno, že nejčastěji využívaný informační systém v České republice je systém Helios. Jeho data ovšem nejsou vhodná pro rozbor.

2.1.1. SAP

Data ze systému SAP jsou uložena ve víceřádkové struktuře. To znamená, že jeden účetní případ se nenachází pouze v jednom řádku v Excelu, ale je rozepsán do více řádků. Zpracování takového záznamu je nezbytné, protože syrová data ze systému SAP jsou velmi těžko čitelná. Ve sloupci Účet HK (Hlavní knihy) se nachází jak účet ze strany Má dáti tak účet ze strany Dal.

V některých případech jsou účetní případy rozděleny do více než tří řádků. Vždy v prvním řádku jsou uvedena pořadová čísla, čísla transakce, datum vyúčtování, číslo dokladu a poznámka k celé účetní operaci. V následujících řádcích jsou pak uvedena čísla účtů hlavní knihy na straně Má dáti a na straně Dal, jejich název, příslušná částka a poznámka ke každé operaci. V některých případech je ve sloupci místo účtu hlavní knihy uveden kód obchodního partnera. Samotná data jsou zobrazena v tabulce 1. Tabulka byla upravena, aby hodnoty v jednotlivých polích byly čitelné.

2. Automatizace postupů při zpracování dat

P. číslo	Číslo transakce	Datum účtování	Série	Č. dokladu	Účet HK /Kód OP	Účet HK /název OP	Má dátí / Dal (FM)	Poznámky
90	95746	2.1.2013	PPSOne	SI 7826724				Příjem materiálu
		2.1.2013			123002	Výrobky PET	296,934	Příjem materiálu
		2.1.2013			613002	Zmena stavu výrobku PET	-70,450	Příjem materiálu
91	95747	2.1.2013	PPSOne	SI 7826725				Příjem materiálu
		2.1.2013			123002	Výrobky PET	651,361	Příjem materiálu
		2.1.2013			613002	Zmena stavu výrobku PET	-523,887	Příjem materiálu
92	95748	2.1.2013	PPSOne	SI 7826726				Příjem materiálu
		2.1.2013			123002	Výrobky PET	228,827	Příjem materiálu
		2.1.2013			613002	Zmena stavu výrobku PET	-98,481	Příjem materiálu

Tabulka 2 - Ukázka účetního deníku jako výstupních dat ze systému SAP Zdroj: Výstup ze systému SAP, soubor od HZ Plzeň spol. s r. o.

2.1.2. K2

Data ze systému K2 jsou uložena v jednořádkové struktuře. To znamená, že na jednom řádku jsou dostupná všechna data týkající se jedné operace, což je pro auditory velmi příznivé. Data v této podobě jsou v takovéto podobě dobře čitelná. Zpracování takového záznamu není nezbytné, ale určité minimalizování přebytečných dat by vhodné být mohlo. K náhledu do struktury slouží tabulka číslo 3.

BA	2013	959	U1	1130000156	13.8.2013	Platba VF 30/2876 KAREA,s.r	221800	311113	30	30	279,2983	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 10/9875 BRNI TO L	221800	311113	10	10	166,7861	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 30/2866 KH-Cetto	221800	311113	30	30	51,38005	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 10/2398 STAVEL PL	221800	311113	10	10	94,64102	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 40/1691 Obec Dobř	221800	311113	40	40	65,83662	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 50/3183 SILNICE K	221800	311113	50	50	71,47241	Kč
BA	2013	959	U1	1130000156	13.8.2013	Platba VF 10/8042 Rentec pr	221800	311113	10	10	56,48302	Kč

Tabulka 3 - Ukázka účetního deníku jako výstupních dat ze systému K2 Zdroj: Výstup ze systému K2, soubor od HZ Plzeň spol. s r. o.

2.1.3. JUMP

Data ze systému JUMP jsou uložena ve víceřádkové struktuře. V případě systému JUMP to znamená, že jeden účetní případ je rozdělen do více řádků. U výstupu ze systému JUMP jsou oproti systému SAP uvedeny účty Má dáti a Dal do jednotlivých sloupců. Jedna účetní operace rozepsána do více řádků je opodstatněná, ale pro účely auditu a pro další možnosti zpracování a filtrování je vhodnější jednořádkový zápis.

Účetní deník pro ZO 1,2,3,4,5,6,7,8 /2013 - Celý deník									

Řádek	Č.-Doklad	Datum	ZO	Částka	Účet MD	Účet Dal	Zak	Text	Vyhotoveno

1	11-000002	3.1.2013	1	459.00		211000		brzd.váleček,řemen-3873	22.2.2013
2	11-000002	3.1.2013	1	79.68	343121			brzd.váleček,řemen-3873	13.3.2013
3	11-000002	3.1.2013	1	379.32	501506		58	brzd.váleček,řemen-3873	22.2.2013
4	21-000001	3.1.2013	1	1910.40	325400	221100		CCS CESKA SPOLECNOST	11.1.2013
5	21-000001	3.1.2013	1	284010.00	321000	221100		FD 1- 861 Doprava a mec	11.1.2013
6	21-000001	3.1.2013	1	7979.00	321000	221100		FD 1- 910 Beton Union P	11.1.2013
7	21-000001	3.1.2013	1	24204.00	321000	221100		FD 1- 895 KOMERSIA WEST	11.1.2013

Tabulka 4 - Ukázka výstupních dat ze systému JUMP. Zdroj: Výstup ze systému JUMP, soubor od HZ Plzeň spol. s r. o.

2.1.4. Vhodný výstup po automatizaci zpracování dat

V předešlých ukázkách vstupních dat bylo zobrazeno několik z vybraných způsobů zápisu dat. V následující tabulce číslo 5 je vyobrazen vhodný výstup po automatizovaném zpracování dat, tedy vhodná podoba dat.

Číslo řádku	Číslo operace	Datum	Má dáti	Dal	Částka	Poznámka	DP	DP
1	00001	2.4.2015	221000	261000	5000	Poznámka 1	#DP	#DP
2	00002	3.4.2015	211000	261000	12000	Poznámka 2	#DP	#DP
3	00003	5.4.2015	613000	123000	50000	Poznámka 3	#DP	#DP
4	00004	8.5.2015	251000	664000	32000	Poznámka 4	#DP	#DP

Tabulka 5 - Ukázka vhodné podoby výstupních dat po automatickém zpracování. Zdroj: Vlastní zpracování

Taková podoba dat je vhodná pro čtení a filtrování. Některé účetní softwary nebo systémy, jako například SAP, mají více vhodných informací v jednom účetním deníku. Proto jsou v posledních dvou sloupcích uvedeny volitelně rozšiřující sloupce Doplnkové pole (DP). Do těchto polí může být vložen například název obchodního partnera.

2.2. Možnosti řešení automatizace

Vstupní data byla představena a rozebrána v minulé kapitole. Tento náhled do struktury a podoby dat byl kritickým pro možnost volby vhodného řešení automatizace zpracování. Následující část práce je věnována vybraným možnostem řešení automatizace a jejich rozbořem. Možností pro zpracování dat a řešení automatizace je velké množství. Každý programovací a skriptovací jazyk pracuje s daty, ale ne všechny programovací jazyky jsou vhodné na tento typ problému.

Jednou z možností řešení automatizace je Visual Basic for Application v prostředí aplikace Microsoft Excel. Ten umožňuje zpracovávat data v Excelu přímo. Další možností řešení je programovací jazyk Java. Poslední z vybraných možností je naprogramování rozšíření modulu účetnictví.

2.2.1. Visual Basic for Applications

Visual Basic for Application (VBA) je programovací jazyk, se kterým se pracuje v prostředí Microsoft Office. Jeho implementace je k nalezení například v MS Word, MS Excel a MS Access. Visual Basic (VB) a VBA mají společný základ. Jádro jazyka VB je součástí instalace MS Office a komunikace s ním probíhá přes různé moduly kódu a grafické uživatelské formuláře. Syntaxe jazyka VBA je pro všechny aplikace stejná. Jediným rozdílem je pouze objektový model aplikace, se kterým pracujeme. Ten má každá aplikace z MS Office jiný. Objektový model aplikace se skládá z objektů, jejich metod (funkcí) a vlastností.

2.2.1.1. Využití

VBA se hodí všude tam, kde je potřeba zautomatizovat, zrychlit, zpřesnit a ulehčit práci. Mnohdy se opakují určité činnosti stále dokola. Takovými činnostmi může být stahování dat, úprava těchto dat, tvorba grafů z nich a rozesílání výstupu dalším uživatelům. Pokud jde o stále stejnou činnost, která se opakuje, tak využití VBA je vhodné. VBA zvládá pracovat bez zásahu člověka a pracuje na pozadí.

2.2.1.2. Proměnné a datové typy

Data, která se do proměnných²¹ ukládají, mohou mít různou podobu a podstatu. Mohou to být celá čísla, záporná čísla, desetinná čísla, seznamy hodnot, datum, text, graf a v případě VBA dokonce celá běžící aplikace Microsoft Word. Proměnné jsou definovány určitým datovým typem. Pro představu jsou v následující tabulce rozebrány jednotlivé datové typy.

Datový typ	Rozsah hodnot	Velikost v paměti
Boolean (logický)	True nebo False	2 byty
Byte	0–255	1 byte
Currency (měnový)	-922337203685477,5808 až 922337203685477,5807	8 bytů
Date (datum)	1.leden 100 až 31.prosinec 9999	8 bytů
Decimal (desetinný)	Maximální hodnota +/- 97,288,162,514,264,337,593,543,950,335 s desetinnou čárkou v libovolném místě	12 bytů
Double (dvojitá přesnost)	-1,79769313486231 až -4,94065645841247E-324 pro záporná čísla, 4,94065645841247E-324 do 1,79769313486232E308 pro kladná	8 bytů
Integer (krátké celé číslo)	-32768 až 32767	2 byty
Long (dlouhé celé číslo)	-2147483648 až 2147483647	4 byty
Object (objekt)	Uložený ukazatel (pointer) na libovolný objekt v paměti	4 byty
Single (jednoduchá přesnost)	-3,402823E38 až -1,401298E-45 pro záporné hodnoty a 1,401298E-45 až 3,402823E38 pro hodnoty kladné	4 byty
String (řetězec)	Může obsahovat až 2 ³¹ znaků, může mít konstantní délku až do přibližně 64 000 znaků	Různá
User defined (uživatelský)	Může obsahovat jeden nebo více různých typů	Různá
Variant	Může obsahovat různé hodnoty a objekty	Různá

Tabulka 6 - Přehled typů proměnných Zdroj: KRÁL, Martin. Excel VBA: výukový kurz. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.

Jedním z nejdůležitějších prvků pro tuto práci je pole. „Pole je indexovaná skupina dat, která se chová jako jedna proměnná. Na jednotlivé prvky pole se odkazujeme celočíselným indexem.“²² Ve složitějších aplikacích a skriptech by se bez využití polí značně komplikovala funkcionality a efektivita. Pole nesmírně usnadňuje práci zejména s databázemi, ale lze ho využít i jinde.

2.2.1.3. Práce s daty

Visual Basic for Application v prostředí aplikace Microsoft Excel pracuje přímo s daty a v uživateli viditelné formě. To znamená, že vstupní data do VBA mohou být přímo zobrazena a dostupná na jednom z listů MS Excelu. Vývojové prostředí pro VBA

²¹ Proměnná je pojmenovaná část paměti, kterou používáme k uložení informace.

²² KRÁL, Martin. Excel VBA: výukový kurz. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.

v MS Excelu se spouští buďto zapnutím panelu pro vývojáře v liště možností nebo klávesovou zkratkou ALT+F11.

Ve vývojovém prostředí VBA je možné vytvářet jednotlivé formuláře a programovat jejich funkce. Toto prostředí je rozděleno podle jednotlivých polí a tlačítek. Je možné programovat přímo do obecné sekce, která obsahuje veškeré metody a funkce nebo je možné programovat do konkrétních sekcí. Při tvorbě tlačítka vzniká vlastní sekce s kódem, který je vykonán, je-li tlačítko stisknuté. Program (nebo skript) je možné poté spustit zeleným „play“ tlačítkem.

2.2.1.4. Shrnutí

Výhodou VBA je rychlá dostupnost dat a efektivita při práci s daty v prostředí aplikace MS Excel. To umožňuje okamžitou kontrolu a bezprostřední ověření funkčnosti kódu. Možnost psát kód přímo ve vývojovém prostředí pro Excel je další neméně důležitou výhodou. Další výhodou jsou nástroje pro úpravu jednotlivých formátů buněk. MS Excel umožňuje následně zpracovávat data bez využití VBA prostřednictvím filtrů a řazení. Tyto nástroje by při využití jiného prostředí, než které nabízí Excel, nemohly být využity. Objektový programovací jazyk Visual Basic for Application je vhodný pro automatizaci zpracování dat a bude využit v této práci pro vytvoření skriptu.

2.2.2. JAVA

Programovací jazyk Java vynalezli v roce 1991 James Gosling, Patrik Naughton, Chris Warth, Ed Frank a Mike Sheridan ze společnosti Sun Microsystems. Tento jazyk se původně nazýval „Oak“, ale v roce 1995 byl přejmenován na „Java“. Hlavní motivací pro vznik Javy byla potřeba vytvoření jazyka nezávislého na platformě, který by se dal použít pro tvorbu softwaru vsazovaného do nejrůznějších zařízení spotřební elektroniky, jako jsou topinkovače, mikrovlnné trouby a dálková ovládání. Problém byl v tom, že většina počítačových jazyků byla (v té době) navržena tak, aby se kompilovala pro určitý cíl. Tím je například C++. ²³

²³ SCHILDT, Herbert. Java 7: výukový kurz. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

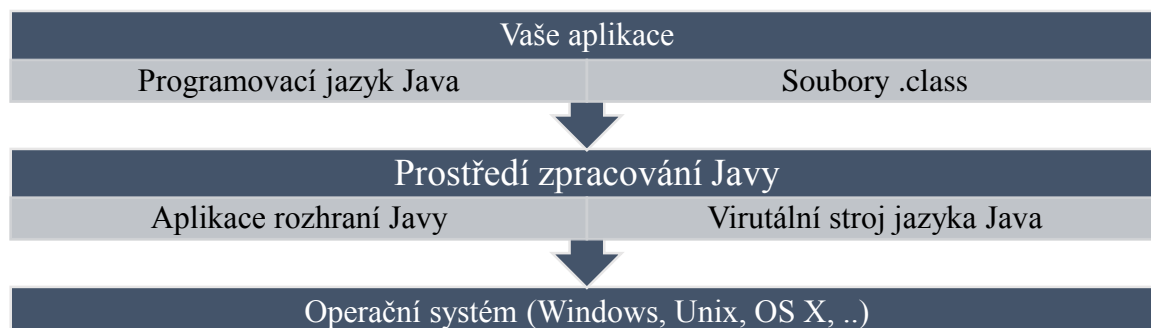
2.2.2.1. Architektura jazyka Java

Celá architektura Javy je kombinací následujících čtyř součástí: programovacího jazyka Java, formátu souboru .class, aplikačního programového rozhraní Javy (API), virtuálního stroje Javy.²⁴

2.2.2.2. Využití (přenositelnost)

Přenositelnost je hlavním aspektem internetu, protože je k němu připojena řada různých typů počítačů a operačních systémů. Pokud by se programy v Javě měly spouštět na prakticky libovolném počítači připojeném k internetu, musel by existovat nějaký způsob, který by umožňoval jejich provádění na různých systémech.

Programy v Javě lze provádět v libovolném prostředí, pro které existuje běhový systém Javy.²⁵ Tuto základní vlastnost Javy je vhodné využít v případě, kdy je zřejmé, že daný program bude spouštěn na různých platformách. Těmi mohou být právě počítače s operačním systémem Windows, počítače od společnosti Apple s operačním systémem OS X nebo počítače běžící s různými distribucemi systému Linux. Prostředí pro zpracování jazyka Java je zobrazeno na obrázku 2.



Obrázek 2 - Prostředí pro zpracování jazyka Java. Zdroj: SPELL, Brett. Java: programujeme profesionálně. Vyd. 1. Překlad Bogdan Kiszka. Praha: Computer Press, 2002, 1022 s. Programujeme profesionálně. ISBN 80-722-6667-5.

2.2.2.3. Proměnné a datové typy

Java obsahuje dvě obecné kategorie datových typů. Jsou jimi objektově orientované a neobjektově orientované. Objektově orientované typy definují třídy. Třída je šablona, která definuje tvar objektu. Stanoví data i kód, který bude na těchto datech pracovat. V následující tabulce jsou vyobrazeny primitivní datové typy jazyka Java.

²⁴ SPELL, Brett. Java: programujeme profesionálně. Vyd. 1. Překlad Bogdan Kiszka. Praha: Computer Press, 2002, 1022 s. Programujeme profesionálně. ISBN 80-722-6667-5.

²⁵ SCHILDT, Herbert. Java 7: výukový kurz. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

Typ	Význam
Boolean	Představují hodnoty pravda a nepravda
Byte	8bitové celé číslo
Char	Znak
Double	Plovoucí desetinná čárka s dvojnásobnou přesností
Float	Plovoucí desetinná čárka s dvojnásobnou přesností
Int	Celé číslo
Long	Dlouhé celé číslo
Short	Krátké celé číslo

Obrázek 3 - Vestavěné primitivní datové typy jazyka Java. Zdroj: SCHILDT, Herbert. Java 7: výukový kurz. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

Dalším datovým typem je pole prvků. Pole prvků se dá indexovat a využívat podobným způsobem jako v jazyce Visual Basic. Práce s poli je důležitá zvláště při zpracování složitějších úloh. Takovou úlohou může být například skript, který by měl na starost zpracování dat.

2.2.2.4. Práce s daty / Práce se soubory

V Javě je k dispozici velká řada tříd a metod, které umožňují čtení a zapisování do souborů. Protože jsou v Javě všechny soubory bajtově orientovány, nabízí Java metody pro čtení a zápis bajtů z a do souboru. Práce se soubory v Javě probíhá prostřednictvím bajtových proudů. Vytvořením těchto proudů je umožněno se soubory pracovat.²⁶

Problémem u těchto proudů je fakt, že práce se soubory probíhá v jiné úrovni než při práci se soubory ve VBA. Otevřením datového proudu je sice možné se soubory pracovat přímo, ovšem ověření správnosti kódu a jeho výstup není tak snadné jako u VBA.

V případě, kdy Java pracuje se soubory, které byly uloženy z prostředí aplikace MS Excel, je jejich zpracování značně zkomplikováno. MS Excel ukládá soubory ve své konkrétní podobě. Samotný soubor z MS Excelu je primárně určen k dalšímu otevření v aplikaci MS Excel. Pokud je tento soubor otevíráný jinou aplikací, může dojít k jeho špatné interpretaci.

Existuje řada nástrojů, které umožňují otevírání souborů z aplikace MS Excel v Javě. Využití těchto nástrojů je ovšem komplikované a ne vždy je dosaženo takového

²⁶ SCHILDT, Herbert. Java 7: výukový kurz. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.

výsledku, jaký byl očekáván. K tomuto závěru jsem došel vlastní analýzou, která ovšem není součástí této práce.

2.2.2.5. Shrnutí

Použití programovacího jazyka pro vytvoření programu, který bude automatizovat zpracování dat, je možné, ovšem je zbytečně složité, protože soubory zpracované a uložené v aplikaci MS Excel se těžko otevírají a zpracovávají. Výhodou Javy je bezpochyby její přenositelnost mezi různé platformy.

2.2.3. Modul do ERP

Poslední vybranou možností zpracování dat je úprava nebo vytvoření vlastního modulu do každého z informačních systémů. Informační systémy, které využívají modul účetnictví, nebo účetní softwary mají každý svůj jistý výstup. Tento výstup se dá často modifikovat, aby splňoval určité požadavky uživatele. Základní možnosti tohoto nastavení nejsou zdaleka tak rozsáhlé, aby výstupy byly snadno čitelné pro auditory. Největším problémem těchto výstupů je ovšem jejich podoba. Je velmi nepravděpodobné, že by se podařilo vytvořit shodný výstup z několika nebo dokonce všech informačních systémů. Možností jak tento problém odstranit je naprogramování skriptu, upravujícího podobu výstupních dat přímo do modulu, který má na starost účetnictví nebo který se stará o výstupy do souboru. Taková varianta řešení by si vyžadovala přístup do zdrojových kódů informačního systému. Vyžadovala by si porozumění celého kódu a celého informačního systému nebo ekonomického či účetního softwaru. Přístup ke zdrojovému kódu by velmi pravděpodobně nebyl umožněn třetí straně, která by se snažila tento kód pozměnit. Každý informační systém může být napsaný v jiném programovacím jazyce, proto by bylo velmi náročné tyto změny provést. V praxi by takové řešení pro auditory znamenalo kontaktování všech svých klientů a úpravu jejich informačních systémů.

2.2.4. Shrnutí možností automatizace

Z vybraných možností řešení automatizace je nejvhodnějším řešením automatizace prostřednictvím programovacího jazyka Visual Basic for Application v prostředí aplikace MS Excel. Takové řešení je vhodné zejména díky vstupním souborům. Těmi jsou soubory v programu MS Excel. Zpracování takových souborů je nejrozmumnější přímo v Excelu. VBA umožňuje sledovat změny přímo a hned po provedení skriptu. VBA je přímo určen pro takový typ úloh a jeho možnosti a nástroje

jsou v tomto ohledu velmi užitečné. Využití programovacího jazyka Java je možné, ale náročné na provedení. Výhody Javy existují, ale jejich využití u takového typu problému nejsou vhodné. Bylo by užitečné mít vytvořený program v Javě a moci ho využít na jakékoli platformě. V dnešní době jsou aplikace MS Office dostupné na všechny větší platformy a tím výhoda Javy v tomto ohledu zaniká. Tvorbou jednotlivých modulů by se dal celý problém eliminovat, ale je velmi nepravděpodobné, že by se výrobci všech informačních systémů dohodli a sjednotili své výstupy. Každý pracuje s daty odlišným způsobem, a proto i jejich výstupy budou vždy rozdílné.

2.3. Návrh a dokumentace skriptu

Na základě analýzy, která byla provedena v předešlé části práce, byl Visual Basic for Application vyhodnocen jako nejvhodnější nástroj pro vytvoření skriptu na zpracování dat. Následující část práce se bude věnovat rozboru a dokumentaci skriptu, který byl napsán ve VBA v prostředí aplikace MS Excel pro účely této bakalářské práce. Tento skript si dává za úkol zjednodušit čtení a orientaci ve výstupech z vybraných účetních deníků. Tento skript zpracovává data do srozumitelnější podoby z výstupů ze systému SAP a JUMP. Tyto dva systémy byly vybrány na základě analýzy.

2.3.1. Systém SAP

2.3.1.1. Analýza vstupních dat

Pro pochopení a možnost popsání skriptu, který byl psán vždy pro konkrétní část účetního deníku, je potřeba rozebrat vstupní data. Tabulka 7 reprezentuje výstupní data z účetního deníku systému SAP.

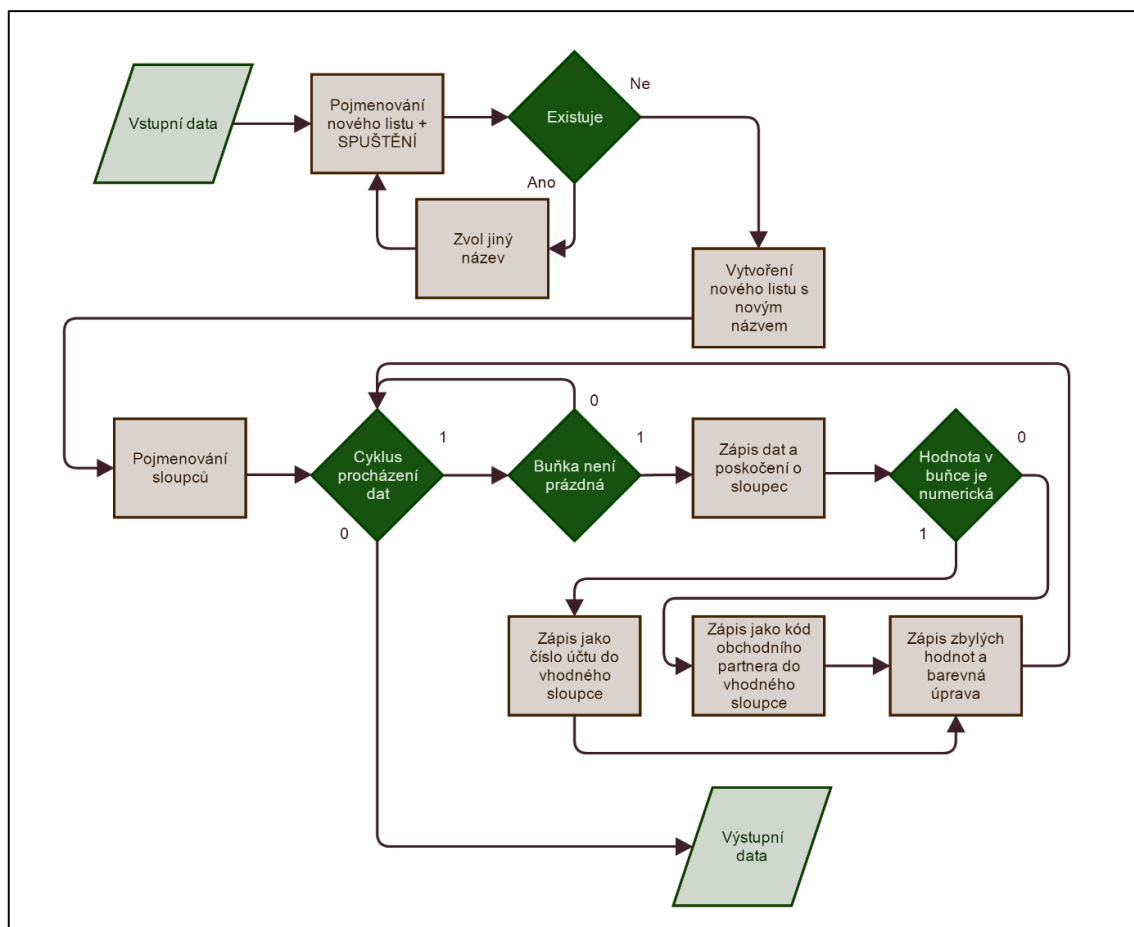
2. Automatizace postupů při zpracování dat

Poř. číslo	Číslo transakce	Datum účtování	Série	Č. dokladu	Účet HK / Kód OP	Účet HK / název OP	Má dáti / Dal (FM)	Poznámky
90	95746	2.1.2013	PPSOne	SI 7826724				Příjem materiálu
		2.1.2013			123002	Výrobky PET	296,9345438	Příjem materiálu
		2.1.2013			613002	Zmena stavu výrobku PET	-70,45084533	Příjem materiálu
91	95747	2.1.2013	PPSOne	SI 7826725				Příjem materiálu
		2.1.2013			123002	Výrobky PET	651,3618958	Příjem materiálu
		2.1.2013			613002	Zmena stavu výrobku PET	-523,8873089	Příjem materiálu
...
104	95760	2.1.2013	PC400	PS 44109164				Odeslané platby - D000216
		2.1.2013			221400	Bankovní účet CS CZK	-40,50206852	Direct Parcel Distribution CZ
		2.1.2013			D000216	Direct Parcel Distribution CZ, s.r.o.	229,3430536	Direct Parcel Distribution CZ, s.r.o.
...
127	95783	2.1.2013	RC500	RC 45100333				UNICREDIT LEASING CZ
		2.1.2013			221500	Bankovní účet CS EUR	119408,4826	UniCredit Leasing CZ, a.s.
		2.1.2013			Z000691	UniCredit Leasing CZ, a.s.	-100571,4829	UniCredit Leasing CZ, a.s.
		2.1.2013			563000	Kurzové ztráty	414,801142	UniCredit Leasing CZ, a.s.

Tabulka 7 - Výtažek z účetního deníku jako výstupních dat ze systému SAP. Zdroj: Výstup ze systému SAP

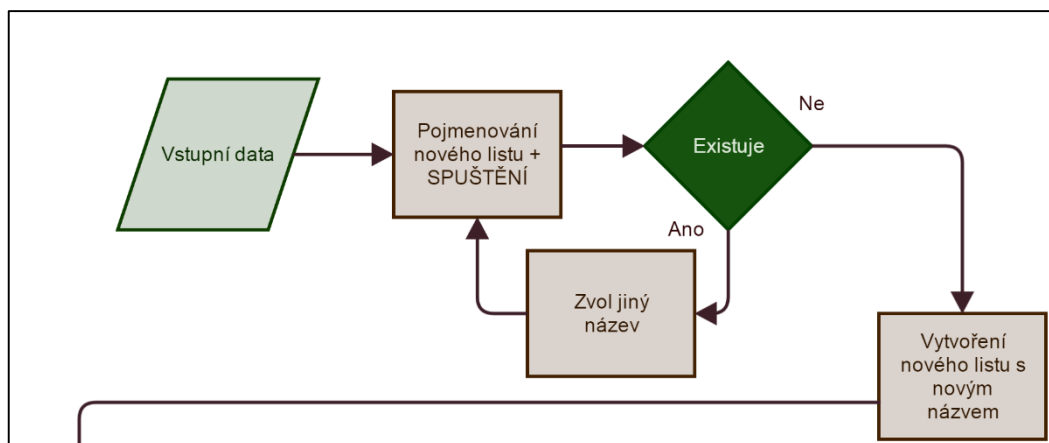
Ve sloupci A se nachází pořadové číslo operace. Pro účely zjednodušení je v předchozí tabulce zjednodušeno a vynecháno několik pořadových čísel, protože u případů s pořadovými čísly 90, 104 a 127 se jedná vždy o jiný způsob zpracování dat. Na tuto tabulku bude v následující části odkazováno. Ve sloupci B se nachází čísla transakce. V sloupci C se nachází datum účtování. Ukázková data mají pouze desítky až stovky řádků a to vše z jednoho data. Ve sloupci D se nachází označení série, které pro zpracování nehraje žádnou významnou roli. V dalším sloupci, ve sloupci E, se nachází číslo dokumentu, které je velmi důležité a pomáhá s orientací. Ve sloupci F se nachází jak účet hlavní knihy v syntetickém i analytickém tvaru tak i kód obchodního partnera. Tento sloupec obsahuje jednu z možností podle typu účetní operace. Ve sloupci G se nachází účet hlavní knihy vyjádřený slovně a název obchodního partnera. Ve sloupci H se nachází částky Má dáti a Dal ve firemní měně. V posledním sloupci I se nachází poznámky k dané účetní operaci.

2.3.1.2. Návrh zpracování dat



Obrázek 4 - Vývojový diagram popisující celý proces zpracování dat ze systému SAP. Zdroj: Vlastní zpracování, obrázek byl vytvořen ve webovém prostředí na serveru gliffy.com

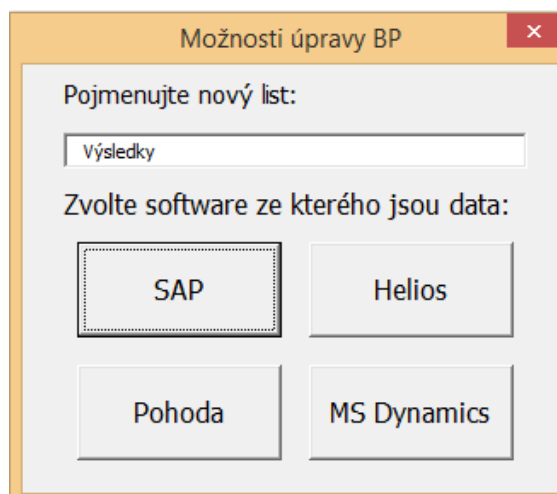
Vývojový diagram na obrázku 4 znázorňuje způsob procházení dat a jejich zápis do nového listu, který je předem definovaný. Tento diagram by měl usnadnit porozumění skriptu, který data zpracovává. Buňky znázorněné jako kosodélník s texty Vstupní data a Výstupní data reprezentují začátek a konec skriptu. Obdélníkové buňky reprezentují jednotlivé procesy. Buňky reprezentovány zelenými čtverci postavenými na rohy symbolizují jednotlivé rozhodující procesy. Těmi mohou být začátky cyklů a různé podmínky. Při jejich splnění se při čtení diagramu pokračuje po větvi, která odpovídá splnění podmínky. To může být označeno jedničkou nebo slovem ano. Pokud podmínka splněná není, pokračuje se po větvi s nulou nebo slovem ne. Následující část bakalářské práce se bude věnovat jednotlivým buňkám a rozboru jednotlivých kroků při zpracování dat a převodu do nového a srozumitelnějšího zápisu.



Obrázek 5 - Vývojový diagram popisující úvodní část procesu zpracování dat ze systému SAP.
Zdroj: Vlastní zpracování, obrázek byl vytvořen ve webovém prostředí na serveru gliffy.com

Buňka Vstupní data znázorňuje vstupování hlavních dat z účetního deníku systému SAP, která byla na přechozích stránkách rozebrána. Vstupní data, která jsou v tomto skriptu využívána, jsou částečným výtažkem z účetnictví jisté společnosti, která zaznamenává všechny základní účetní operace. Proto jsou ve vstupních datech k nalezení položky s odesláním platby či například příjem materiálu.

Buňka s pojmenováním nového listu a spuštěním je referencí ke grafickému uživatelskému prostředí. V tomto prostředí je v prvním kroku pojmenováván nový list, na který bude celkový výstup zapsán a uložen.



Obrázek 6 - Ukázka grafického uživatelského prostředí pro ovládání skriptu. Zdroj: Vlastní zpracování

Na obrázku 6 je vyobrazeno grafické prostředí, které dává uživateli možnost pojmenovat nový list, ve kterém je již předdefinován název „Výsledky“. Hned poté uživatel zvolí účetní software nebo informační systém, ze kterého jsou vstupní data. Tento bod je kritický, protože za každým tlačítkem se skrývá jiný algoritmus, který pracuje

naprosto odlišně a nebyl by schopný zpracovat data z jiného účetního deníku než z toho, pro který je určen. Obrázek 6 je první verzí grafické rozhraní. Ve finální verzi je i systém JUMP.

```

Dim nazevListu As String

nazevListu = TextBox1.Text
rowCount = List1.UsedRange.Rows.Count

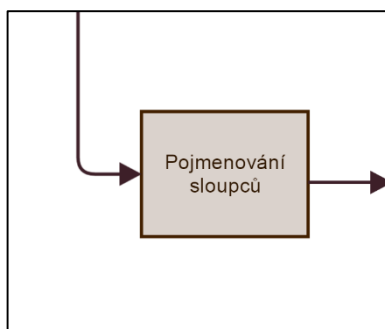
Dim ws As Worksheet

Set ws =
ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.Count))
ws.Name = nazevListu

```

Obrázek 7 - Část kódu SAP #1. Zdroj: Vlastní zpracování

Na obrázku 7 s je zobrazena část skriptu, která má na starost přečtení a definování proměnné nazevListu a následné definování proměnné ws (worksheet), do které je tento název uložen. V prvním řádku je deklarována proměnná nazevListu jako řetězec String. Tato proměnná je definována obsahem textového pole TextBox1. Je deklarována proměnná ws jako Worksheet a je nastavena tak, aby nový list byl umístěn až za poslední list současného souboru podle počtu existujících listů. V posledním kroku je definován název objektu ws z hodnoty obsažené v proměnné nazevListu. Je zde i definována proměnná rowCount, který obsahuje počet řádků.



Obrázek 8 - Vývojový diagram zobrazující proces pojmenování sloupců při zpracování dat ze systému SAP. Zdroj: Vlastní zpracování

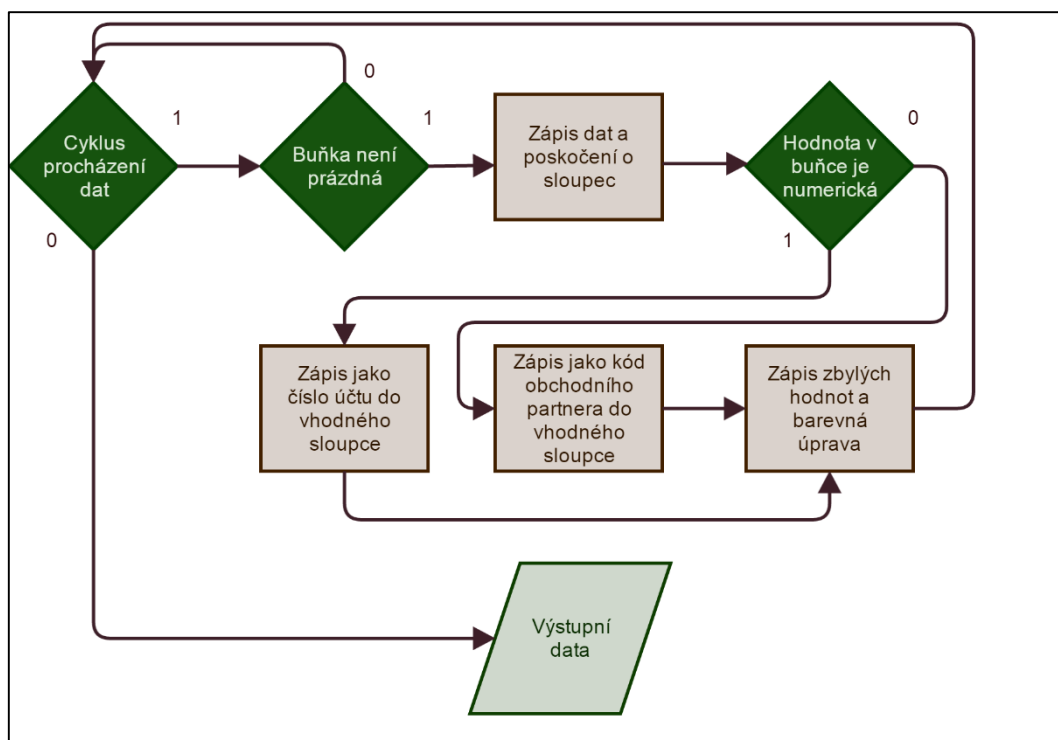
Buňka Pojmenování sloupců (obrázek 8) reprezentuje část skriptu, kde vzniká nový list a kde jsou pojmenovány sloupce na novém listě. Tyto hodnoty nových sloupců jsou již prvním krokem k zjednodušení čtení původních dat. Rozmístění těchto jednotlivých názvů je snadno konfigurovatelné pro potřeby uživatele. Zásadní v tomto pojmenování je rozdělení položek Účet HK/Kód OP na několik sloupců. V původních datech byly tyto položky všechny v jednom stejném sloupci a to nebylo vhodné

pro srozumitelné čtení. Tento sloupec bude rozdělen do tří sloupců, ve kterém bude každý sloupec reprezentovat jiná data. V jednom ze sloupců bude uvedeno účtování na straně Má dáti, v druhém bude zapsáno účtování na straně Dal a v posledním sloupci bude uveden Kód partnera. Tímto rozdělením se usnadní čtení a orientace v tabulce.

```
ws.Range("A1").Value = "Číslo operace"
ws.Range("B1").Value = "Datum operace"
ws.Range("C1").Value = "Číslo dokladu"
ws.Range("D1").Value = "Má Dáti"
ws.Range("E1").Value = "MD/D (FM)"
ws.Range("F1").Value = "Dal"
ws.Range("G1").Value = "MD/D (FM)"
ws.Range("H1").Value = "Kód partnera"
ws.Range("I1").Value = "Název partnera"
ws.Range("J1").Value = "MD/D (FM)"
ws.Range("K1").Value = "Poznámka"
ws.Rows(1).Interior.Color = 204255255
```

Obrázek 9 - Část kódu SAP #2. Zdroj: Vlastní zpracování

Do již předem deklarovaného a definovaného objektu ws jsou do pole Range, který ukazuje na určitou buňku v objektu ws, zapisovány jednotlivé hodnoty buňek. Bylo opuštěno pojmenování sloupců ze vstupních dat Účet HK (hlavní knihy) a byly nahrazeny dvěma sloupci Má dáti a Dal, které u sebe vždy obsahují hodnotu z patřičné účetní operace.



Obrázek 10 - Vývojový diagram popisující hlavní proces zpracování dat ze systému SAP. Zdroj: Vlastní zpracování

Na obrázku 10 je zobrazena hlavní část skriptu, která řeší celé zpracování dat ze systému SAP. V této části je hlavní cyklus, který prochází první sloupec vstupních dat až do celkového počtu řádků, který byl na začátku definován. Dalším krokem je posouzení, zda-li je sledovaná buňka prvního sloupce prázdná či nikoliv a pokud ano, pokračuje cyklus na další buňku, u které probíhá stejné ověření. Tohle ověření je zde z toho důvodu, že ne každý záznam ve vstupních datech obsahuje stejný počet řádků. To je zjištělné z tabulky č. 3. Operace číslo 90 obsahuje informace o účtování na stranu má dáti i na stranu dal. Operace číslo 127 obsahuje informace o účtování na stranu má dáti, označení obchodní partnera kódem a informace o účtování na straně dal. Účetní operace typu Pořízení materiálu je zapsána na třech řádcích, a ostatní operace kolikrát na víc než třech. Proto nejde použít cyklus, který by pouze přehazoval data podle řádků, ale musí být použit cyklus, který vyhledává neprázdné buňky a dále pracuje s daty tak, jak jsou zapsána. Pokud kontrolovaná buňka prvního sloupce prázdná není a obsahuje skutečně číslo transakce, může skript pokračovat a sledovat další buňky. Dojde k zápisu dat z prvního sloupce a zahájí se kontrola sloupce F, který obsahuje účet hlavní knihy nebo kód obchodního partnera. Kódy obchodních partnerů začínají písmenem D, tedy obsahem buňky není číslo, nýbrž textový řetězec. Účty hlavní knihy jsou číselné. První tři písmena reprezentují syntetický účet a následující čísla účet analytický. Těchto šest čísel je v buňce reprezentováno šesticiferným číslem, nikoliv textovým řetězcem. V tuto chvíli může být sledovaná buňka prohlédnuta a může být rozhodnuto o tom, zda se jedná o číslo nebo textový řetězec. V případě že se jedná o textový řetězec, je hodnota sledované buňky zapsána do vhodného sloupce a skript pokračuje k zápisu dalších hodnot do dalších vhodných sloupců. Pokud ovšem hodnota numerická je, tak skript pochopí, že se jedná o účtování a o účet hlavní knihy a hodnoty zapisuje do jiných, vhodnějších sloupců. V tom případě, kdy je hodnota numerická, skript automaticky vyhledá i stranu dal a zapisuje hodnoty do příslušných sloupců. Tento cyklus se opakuje do konce celého dokumentu a výstupem je zpracovaná tabulka na novém již pojmenovaném listě.

```

Dim i As Integer, radek As Integer
radek = 2
For i = 2 To rowsCount
    If Not IsEmpty(List1.Cells(i, 2)) Then
        If radek Mod 2 = 1 Then
            ws.Rows(radek).Interior.Color = &HD8D8D8
        Else
            ws.Rows(radek).Interior.Color = &HE8E8E8
        End If
        ws.Cells(radek, 1) = List1.Cells(i, 2)
        ws.Cells(radek, 2) = Format(List1.Cells(i, 3), "dd/mm/yyyy")
        ws.Cells(radek, 3) = List1.Cells(i, 5)
        Dim j As Integer, MDD As Integer
        j = 1
        MDD = 0
        While Not IsEmpty(List1.Cells(i + j, 6))
            If IsNumeric(List1.Cells(i + j, 6)) Then
                If MDD = 0 Then
                    ws.Cells(radek, 4) = List1.Cells(i + j, 6)
                    ws.Cells(radek, 5) = List1.Cells(i + j, 8)
                    MDD = 1
                ElseIf MDD = 1 Then
                    ws.Cells(radek, 6) = List1.Cells(i + j, 6)
                    ws.Cells(radek, 7) = List1.Cells(i + j, 8)
                End If
            ElseIf Not IsNumeric(List1.Cells(i + j, 6)) Then
                ws.Cells(radek, 8) = List1.Cells(i + j, 6)
                ws.Cells(radek, 9) = List1.Cells(i + j, 7)
                ws.Cells(radek, 10) = List1.Cells(i + j, 8)
            End If
            j = j + 1
        Wend
        ws.Cells(radek, 11) = List1.Cells(i, 9)
        radek = radek + 1
    End If
Next i

```

Obrázek 11 - Část kódu SAP #3. Zdroj: Vlastní zpracování

Na obrázku 11 je zobrazen kód, reprezentující jednotlivé cykly a podmínky pro vykonání celého problému. Tento kód je potřeba rozdělit do několika oddílů a podrobně je rozepsat.

```

Dim i As Integer, radek As Integer
radek = 2
For i = 2 To rowsCount

```

Obrázek 13 - Část kódu SAP #4. Zdroj: Vlastní zpracování

Na obrázku 13 části kódu Dim i deklaruje proměnnou i jako integer. Proměnná i je jednou z kritických proměnných, protože slouží jako ukazatel na jednotlivé buňky a zároveň je porovnávána s proměnnou rowsCount, neboli s počtem řádků. Na posledním řádku začíná cyklus for, který běží dokud zmíněná proměnná i nedosáhne počtu řádků v celém dokumentu. Její hodnota je ze začátku definována jako 2, protože na prvním řádku jsou zapsané popisy jednotlivých sloupců.

```

If Not IsEmpty(List1.Cells(i, 2)) Then
    If radek Mod 2 = 1 Then
        ws.Rows(radek).Interior.Color = &HD8D8D8
    Else
        ws.Rows(radek).Interior.Color = &HE8E8E8
    End If

```

Obrázek 12 - Část kódu SAP #5. Zdroj: Vlastní zpracování

IsEmpty() je funkcí, která zjišťuje, zdali je zadaná buňka v parametru funkce prázdná nebo nikoliv. Pokud zadaná buňka v parametru funkce prázdná je, funkce vrátí true. Aby mohla být podmínka splněná znegovaně, je potřeba před funkcí IsEmpty() vložit negující operátor Not, který zajistí, aby podmínka If byla splněna jen tehdy, pokud buňka v parametru funkce IsEmpty() prázdná není. If radek Mod 2 = 1 posuzuje, zdali je hodnota uložená v proměnné radek sudá nebo lichá a na základě toho obarvuje jednotlivé řádky jistou barvou. Každý sudý a lichý řádek se od sebe barevně liší v drobných odstínech šedi.

```

ws.Cells(radek, 1) = List1.Cells(i, 2)
ws.Cells(radek, 2) = Format(List1.Cells(i, 3), "dd/mm/yyyy")
ws.Cells(radek, 3) = List1.Cells(i, 5)
Dim j As Integer, MDD As Integer
j = 1
MDD = 0

```

Obrázek 14 - Část kódu SAP #6. Zdroj: Vlastní zpracování

Objekt ws je objektem nového listu. Prostřednictvím pole Cells je možné ukládat hodnoty do jednotlivých buněk tohoto listu. První tři řádky kódu zapisují do jednotlivých sloupců data z prvního listu dokumentu. V prvním sloupci jsou hodnoty z druhého

sloupce neboli čísla operace. V druhém sloupci je datum provedení operace, které je přeformátováno funkcí Format(), která zajistí, aby data byla v čitelné formě a aby nebyla vyjádřena číslicí. V třetím sloupci je číslo dokladu, které je pouze překopírováno z prvního listu dokumentu.

Následně je deklarována proměnná *j*, která je typu integer, pro procházení jednotlivých sloupců a řádků v prvním listu. Tato proměnná společně s proměnnou *i* slouží ve skriptu k ukazování na jednotlivé buňky. Procházení těchto buněk je totožné s procházením dvourozměrných polí v jiných programovacích jazycích, jako je například Java nebo PHP. Pod proměnnou *i* se tedy skrývá první rozměr, kterým je řádek, a pod proměnnou *j* se skrývá přidaná část k prvnímu rozměru, který pomáhá ukázat na následující řádek. Dále je zde deklarována proměnná MDD, která zastupuje Má dáti/Dal. Ta slouží jako pomocná proměnná, která nabývá hodnot na základě posouzení, zdali se jedná o účtování na stranu Má dáti nebo o účtování na stranu Dal. Obě tyto proměnné jsou definovány. Proměnná *j* je definována jedničkou a proměnná MDD je definována nulou.

```
While Not IsEmpty(List1.Cells(i + j, 6))
  If IsNumeric(List1.Cells(i + j, 6)) Then
    If MDD = 0 Then
      ws.Cells(radek, 4) = List1.Cells(i + j, 6)
      ws.Cells(radek, 5) = List1.Cells(i + j, 8)
      MDD = 1
    ElseIf MDD = 1 Then
      ws.Cells(radek, 6) = List1.Cells(i + j, 6)
      ws.Cells(radek, 7) = List1.Cells(i + j, 8)
    End If
    ElseIf Not IsNumeric(List1.Cells(i + j, 6)) Then
      ws.Cells(radek, 8) = List1.Cells(i + j, 6)
      ws.Cells(radek, 9) = List1.Cells(i + j, 7)
      ws.Cells(radek, 10) = List1.Cells(i + j, 8)
    End If
    j = j + 1
  Wend
```

Obrázek 15 - Část kódu SAP #7. Zdroj: Vlastní zpracování

Kód na obrázku 15 se zabývá procházením jednotlivých buněk v již předem schváleném řádku. Jedná se o cyklus s podmínkou na začátku s předem neurčeným počtem opakování. Podmínkou pro tento cyklus neprázdná buňka, na kterou je ukazováno proměnnou *i* a proměnnou *j*.

Podmínka `if IsNumeric()` je splněna, pokud funkce `IsNumeric()` vrátí `true`. `true` vrátí v případě, kdy sledovaná buňka je číselná. Pokud číselná není, je znovu posuzována, zdali není nečíselnou a pokud ano, je prováděna spodní část kódu. Pokud číselná je, přichází na řadu podmínka, která je splněná, jeli proměnná MDD nulová. Pokud nulová je, jedná se o stranu Má dáti a skript začne zapisovat jednotlivé obsahy buněk na stranu Má dáti. Následně je tato proměnná MDD změněna na jedničku a při příštím procházení cyklem je splněna podmínka druhá. Při splnění této podmínky se jedná o účtování na stranu Dal a skript začne zapisovat jednotlivé obsahy buněk na stranu Dal. V této části kódu není potřeba opět proměnnou MDD změnit na nulu, protože v každé účetní operaci se vyskytuje pouze jedenkrát strana Má dáti a jedenkrát strana Dal. Pokud by tomu tak nebylo jako například u složitějších operací, je možné tuto proměnnou upravit do podoby, kdy její hodnota bude nabývat jiných hodnot, aby byla splněna potřeba uživatele.

Možnou variantou by mohlo být přepisování této proměnné vždy po splnění podmínky, dokud se jedná o stejnou stranu účtování. V případě, že by se například účtovalo s daní z přidané hodnoty je možné tuto proměnnou pozdržet. Pozdržením proměnné by bylo docíleno zopakování stejné podmínky a opětovné účtování na stranu dal. Pro dokončení příkladu s daní z přidané hodnoty jsou přiloženy obrázky 16 a 17, které tohle vysvětlují výsledek.

Má dáti	Dal
311.000	
	601.000
	343.000

Obrázek 17 - Stav A

Má dáti	Dal
311.000	601.000
311.000	343.000

Obrázek 16 - Stav B

V poslední řadě je v kódu část, která ověřuje, jestli je sledovaná buňka nečíselného typu. Pokud tato buňka nečíselná je, zapisujeme hodnoty z oblasti do buněk jako kód a název obchodního partnera společně s komentářem z posledního sloupce. Před koncem cyklu je proměnná `j` ještě zvýšena o jednotku, aby při příštím procházení cyklem bylo ukazováno na řádek o jeden vyšší.

2.3.1.3. Shrnutí

V předchozí části byla analyzována vstupní data a navrženo zpracování těchto dat. Tento způsob zpracování spoléhá na to, že data jsou přesně ve stejné struktuře, v jaké byly pro tuto práci k dispozici. V některých sloupcích výstupních dat ze systému SAP se nachází několik odlišných hodnot a je potřeba tyto hodnoty od sebe odlišit. Z toho důvodu

2. Automatizace postupů při zpracování dat

je několikrát použita funkce zjišťující, jestli se jedná o číselnou hodnotu a jestli je sledovaná buňka prázdná. Tímto způsobem se dají procházet a zpracovávat data, která jsou zapsána do podobných bloků. Úpravou kódu se dá docílit zpracování dat z jakéhokoli informačního systému, který má výstup v podobné struktuře.

2.3.1.4. Výstup

Po zpracování dat a obarvení příslušných řádků je výstup čitelnější než původní data. Celkový výstup po zpracování dat ze systému sap skriptem je na obrázku 18

Číslo operace	Datum operace	Číslo dokladu	Má Dátu	MD/D (FM)	Dal	MD/D (FM)	Kód partnera	Název partnera	MD/D (FM)	Poznámka
95746 02. 01. 2013	SI 7826724	123002	296,9545438	613002	-70,45084533					Přijem materiálu
95747 02. 01. 2013	SI 7826725	123002	651,3618958	613002	-523,8873089					Přijem materiálu
95748 02. 01. 2013	SI 7826726	123002	228,8276093	613002	-98,48186643					Přijem materiálu
95749 02. 01. 2013	SI 7826727	123002	244,628588	613002	-413,1736612					Přijem materiálu
95750 02. 01. 2013	SI 7826728	123002	404,6265637	613002	-36,89022824					Přijem materiálu
95751 03. 01. 2013	SI 7826729	123002	136,8753963	613002	-73,9332046					Přijem materiálu
95752 02. 01. 2013	PS 44109156	221400	-1,404152219			D000027	Miroslav Moller		0,830208516	Odeslané platby - D000027
95753 02. 01. 2013	PS 44109157	221400	-45,75314636			D000060	INVEST TEL, s.r.o.		306,000991	Odeslané platby - D000060
95754 02. 01. 2013	PS 44109158	221400	-75,94881586			D000150	V. Cenek za sdružení SEKAT		36,04380687	Odeslané platby - D000150
95755 02. 01. 2013	PS 44109159	221400	-414,7663943			D000230	ELIOD servis, s.r.o.		376,3150342	Odeslané platby - D000230
95756 02. 01. 2013	PS 44109160	221400	-944,9794126			D000392	Reitmajer Dušan-Truhlářství		559,6586712	Odeslané platby - D000392
95757 02. 01. 2013	PS 44109161	221400	-265,0686018			D000624	Trend Práce s.r.o.		1202,19891	Odeslané platby - D000624
95758 02. 01. 2013	PS 44109162	221400	-7,170841159			D000022	Linde Gas a.s.		2,682907716	Odeslané platby - D000022
95759 02. 01. 2013	PS 44109163	221400	-208,3789661			D000093	Jiri Altman - truhlářství		1532,673848	Odeslané platby - D000093
95760 02. 01. 2013	PS 44109164	221400	-40,50206852			D000216	Direct Parcel Distribution CZ, s.r.o.		229,3430536	Odeslané platby - D000216
95761 02. 01. 2013	PS 44109165	221400	-1223,476114			D000404	Dineos s.r.o.		1374,903094	Odeslané platby - D000404
95762 02. 01. 2013	PS 44109166	221400	-2290,001756			D000623	DU-MI CZ, a.s.		1242,828707	Odeslané platby - D000623
95763 02. 01. 2013	PS 44109167	221400	-1574,52489			Z000599	NYPACK, s.r.o.		1894,62061	Odeslané platby - Z000599
95764 02. 01. 2013	PS 44109168	221400	-744,5916593			D000357	TOPTRANS EU a.s. org.složka		1026,108311	Odeslané platby - D000357
95765 02. 01. 2013	PS 44109169	221400	-178,6592526			Z000599	NYPACK, s.r.o.		561,285278	Odeslané platby - Z000599
95766 02. 01. 2013	PS 44109170	221400	-647,2930029			D000052	Mgr. JINDRICH SOJKA - ADVOKÁT		932,1238006	Odeslané platby - D000052
95767 02. 01. 2013	PS 44109171	221400	-364,2260885			D000155	VALENTA PAVEL		368,4282717	Odeslané platby - D000155
95768 02. 01. 2013	PS 44109172	221400	-222,5010234			D000610	CMC Graduate School of Business o.p.s		535,5873217	Odeslané platby - D000610
95769 02. 01. 2013	PS 44109173	221400	-734,8009394			D000313	GERLACH spol. s.r.o.		682,2305929	Odeslané platby - D000313
95770 02. 01. 2013	PS 44109174	221400	-590,4564552			D000304	Generali Pojišťovna a.s.		93,22492843	Odeslané platby - D000304
95771 02. 01. 2013	PS 44109175	221400	-98,80473761			D000328	Josef Benda Autodoprava		218,093326	Odeslané platby - D000328
95772 02. 01. 2013	PS 44109176	221400	-94,0522494			D000209	DEVELOP-start spol. s.r.o.		208,3940651	Odeslané platby - D000209
95773 02. 01. 2013	PS 44109177	221400	-65,97089256			D000104	Josef Aichinger		2,800899351	Odeslané platby - D000104
95774 02. 01. 2013	PS 44109178	221400	-156,0792245			D000628	INITIAL ECOTEX s.r.o.		112,0496326	Odeslané platby - D000628
95775 02. 01. 2013	PS 44109179	221400	-19,78884985			D000428	PRIMAGAS s.r.o.		29,31271849	Odeslané platby - D000428
95776 02. 01. 2013	PS 44109180	221400	-28,63373927			D000428	PRIMAGAS s.r.o.		24,32543423	Odeslané platby - D000428
95777 02. 01. 2013	PS 44109181	221400	-69,42715916			D000133	WIMA plast, s.r.o.		55,06803914	Odeslané platby - D000133
95778 02. 01. 2013	PS 44109182	221400	-38,8804622			D000358	Marie Darková		21,96919661	Odeslané platby - D000358
95779 02. 01. 2013	RC 44101546	221400	2064,30249			Z000170	PARENTAL A.S.		-108,8023141	PARENTAL A.S.
95780 02. 01. 2013	PS 44109183	221400	-0,193394928	568000	0,403967903					potovné za přehled
95781 02. 01. 2013	PS 45105534	221500	-62,28318605	568000	19,98462271					POPLATKY FUNDS TRANS

Obrázek 18 - Výstup po zpracování dat ze systému SAP skriptem. Zdroj: Vlastní zpracování

2.3.2. Systém JUMP

2.3.2.1. Analýza vstupních dat

Výstupní data z účetního deníku systému JUMP se výrazně liší od výstupů ze systému SAP. Ve výstupu z účetního deníku systému SAP byly prvky rozloženy do jednotlivých bloků, které začínaly a končily s příslušným označením. Data ze systému JUMP jsou odlišná hlavně tím, že jejich struktura se blíží řádkovému zápisu, který je lépe čitelný. Ovšem i tato data potřebují zpracovat a po dokončení skriptu, který to obstará, bylo zjištěno, že data ze systému JUMP jsou náročnější na zpracování než data ze systému SAP.

A	B	C	D	E	F	G	H	I	J
Řádek	Č. dokladu	Datum	ZO	Částka	Účet MD	Účet Dal	Zak	Text	Vyhotoveno

Tabulka 8 - Hlavička výstupních dat ze systému JUMP. Zdroj: Výstup ze systému JUMP

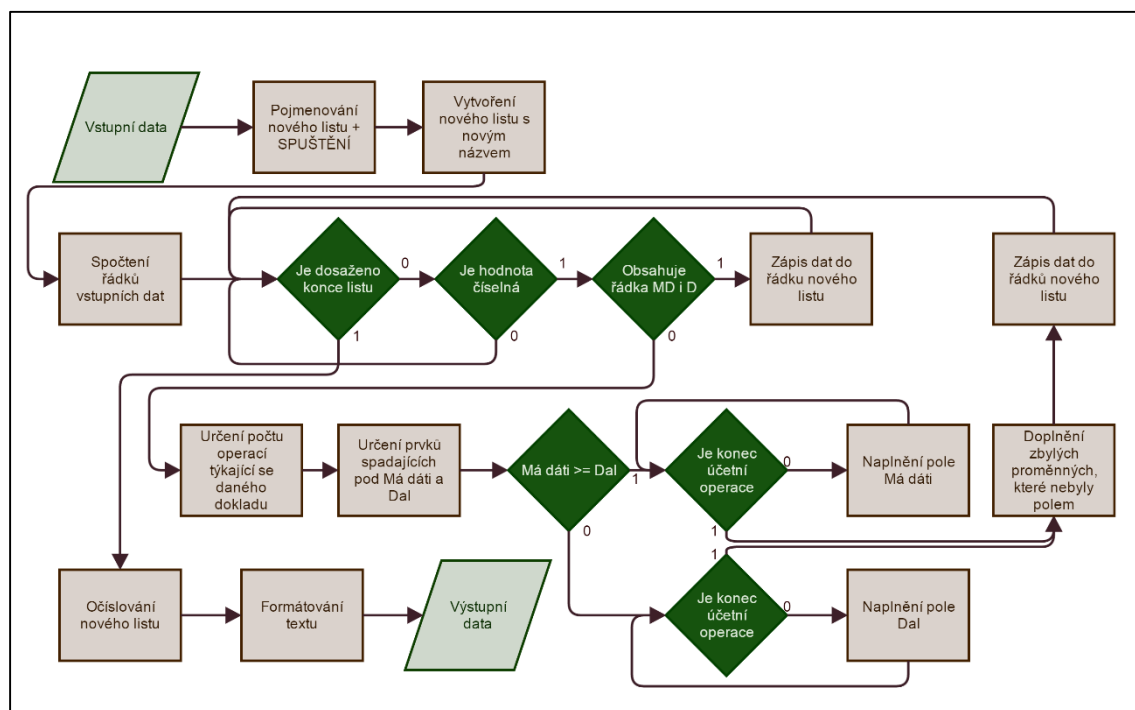
Tabulka číslo 4 je kopií hlavičky účetního deníku systému JUMP. Pro účely skriptu jsou důležité sloupce A, B, C, E, F, G a I. Ve sloupci A se nachází číslování řádků, které pomáhá navigaci při procházení vstupními daty do skriptu. Ve sloupci B se nachází číslo dokladu, které je v tomto zpracování dat klíčové, protože číslo dokladu může skript rozpoznat a určit kolik řádků připadá dané účetní operaci a do kolika řádků na novém listu bude tato operace zaznamenána. Ve sloupci C se nachází datum. Ve sloupci E se nachází částka. Tato částka je v řádcích s jednou operací na doklad i v řádcích s více operacemi k jednomu dokladu. V případě, že je více operací k jednomu dokladu, je částka rozdělena podle příslušných účtů. Ve sloupcích F a G se nachází účet Má dáti a účet Dal. Do těchto sloupců je zapisováno na obě strany, pokud se jedná o operaci, která je jednoduchá. Složitější operace jsou zapsány do více řádků a nevyskytují se proti sobě účetní strany Má dáti a Dal. Posledním řádkem důležitým pro skript a zjednodušení čtení je sloupec I, ve kterém se nachází text popisující operaci.

A	B	C	E	F	G	I
Řádek	Č.-Doklad	Datum	Částka	Účet MD	Účet Dal	Text
1	11-000002	3.1.2013	459.00		211000	brzd.váleček,řemen-3873
2	11-000002	3.1.2013	79,68	343121		brzd.váleček,řemen-3873
3	11-000002	3.1.2013	379.32	501506		brzd.váleček,řemen-3873
4	21-000001	3.1.2013	1910.40	325400	221100	CCS CESKA SPOLECNOST
5	21-000001	3.1.2013	284010.00	321000	221100	FD 1- 861 Doprava a mec
6	21-000001	3.1.2013	7979.00	321000	221100	FD 1- 910 Beton Union P

Tabulka 9 - Vstupní data ve zjednodušené a relevantní podobě. Zdroj: Výstup ze systému JUMP / Vlastní zpracování

V tabulce 9 je zobrazena zjednodušená podoba dat, se kterými bude skript pracovat. Tento zjednodušený tvar dat zobrazuje několik situací. V řádku číslo 1 je číslo dokladu stejné jako u řádku 2 a 3 a velikost částky v řádku číslo 1 je sumou částek řádků 2 a 3. Takový styl zápisu je pro lidské oko velmi přirozená a srozumitelná. Pro Visual Basic, nebo počítač obecně, tomu tak není. Aby mohl data pochopit tak, jak je vidí člověk, je potřeba mu jednotlivé položky vysvětlit a správně s nimi pracovat. V řádku číslo 4 se nachází stejné číslo dokladu jako v řádku 5 a 6 a přitom se jedná o jinou účetní operaci. Jedná se pravděpodobně o stejný doklad, protože je to účtování s bankou (221100) a dokladem je nejspíš výpis z bankovního účtu. Takový doklad obsahuje několik transakcí, které se v účetnictví projeví v několika řádcích a v odlišných operacích. Protože je ovšem číslo dokladu stejné u těchto operací, nemůže číslo dokladu sloužit jako jednoznačný identifikátor operace, proto je potřeba porovnávat tento částečný identifikátor s výskytem účtů Má dáti a Dal. Pokud je v řádku uveden jak Účet MD i Účet Dal, tak se jedná o samostatnou účetní operaci.

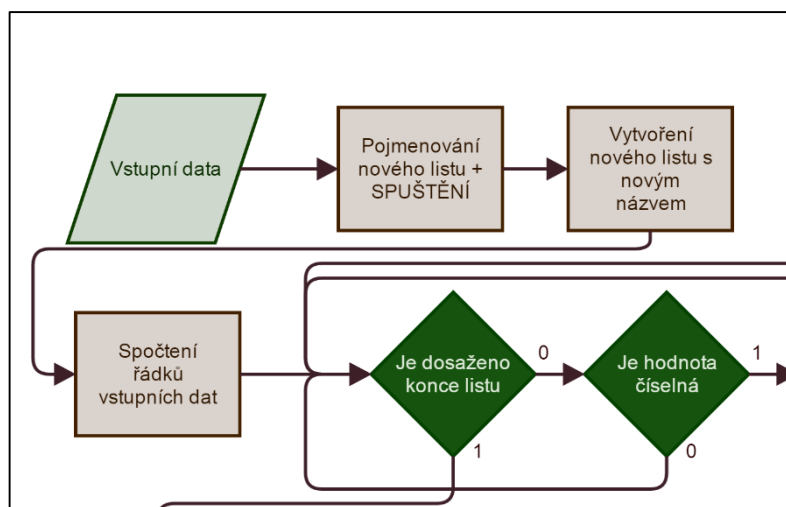
2.3.2.2. Návrh zpracování dat



Obrázek 19 - Vývojový diagram popisující celý proces zpracování dat ze systému JUMP. Zdroj: Vlastní zpracování

Na obrázku 19 je zobrazen vývojový diagram, který popisuje celý proces zpracování dat ze systému JUMP. V úvodní části diagramu se vyskytuje téměř totožný proces jako u systému SAP, který má na starost vytvoření nového listu na základě jména,

které mu dá uživatel. Tímto vytvořením a volbou o jaký systém se jedná (viz obrázek číslo 7) se celý skript spustí a je ukončen až po vytvoření nového výstupu.



Obrázek 20 - Vývojový diagram popisující úvodní procesy zpracování dat ze systému JUMP. Zdroj: Vlastní zpracování.

Spuštěním celého skriptu skrývajícím se po tlačítkem JUMP se spustí první proces, který vytvoří nový list.

```

Set ws =
ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.Count))
ws.Name = nazevListu
  
```

Obrázek 21 - Část kódu JUMP #1. Zdroj: Vlastní zpracování

Tento nový list je hned pojmenován a do první řádky jsou zapsány jednotlivé názvy sloupců. V posledním kroku dojde k obarvení prvního řádku, aby byl graficky odlišen od ostatních.

```

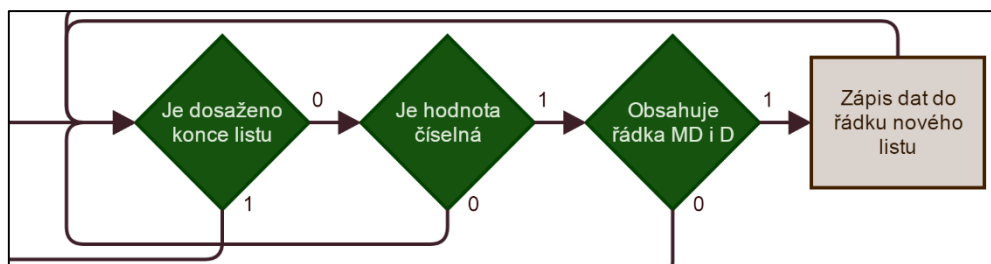
ws.Range("A1").Value = "Číslo řádku"
ws.Range("B1").Value = "Datum operace"
ws.Range("C1").Value = "Číslo dokladu"
ws.Range("D1").Value = "Má Dáti"
ws.Range("E1").Value = "Dal"
ws.Range("F1").Value = "Částka"
ws.Range("G1").Value = "Poznámka"
ws.Rows(1).Interior.Color = 204255255
  
```

Obrázek 22 - Část kódu JUMP #2. Zdroj: Vlastní zpracování

Je patrné, že nejsou využívány všechny hodnoty ze vstupních, ale jen ty, o kterých bylo psáno v části Analýza vstupních dat. Tyto data jsou stěžejní pro účely splnění zadání. V okamžik, kdy je základní formátování a úprava nového listu hotová, je spuštěn hlavní

část zpracování dat, kterou je procházení všech záznamů, jejich rozbor a zápis do nového srozumitelnějšího zápisu.

Hlavní část procházení dat je cyklus s několika vnořenými cykly, který nahlíží do jednotlivých buněk. Tento cyklus závisí na počtu řádek ve vstupních datech a na určení, zdali je sledovaná hodnota číselná.



Obrázek 23 - Vývojový diagram popisující jednu z větví procesu zpracování dat ze systému JUMP. Zdroj: Vlastní zpracování

V části s analýzou vstupních dat bylo zmíněno několik situací, které mohou nastat. První a tou nejzákladnější situací může být stav, kdy je v jednom řádku číslo dokladu, částka, účet má dáti i účet dal. Takový stav je ideální a může být automaticky zapsán, bez dalšího ověření nebo zkoumání. Tento stav je vyobrazen na obrázku 23.

```

For i = 7 To rowsCount

    ws.Cells(radek, 1) = radek - 1

    If IsNumeric(List1.Cells(i, 1)) And Not IsEmpty(List1.Cells(i, 1)) Then

        If Not IsEmpty(List1.Cells(i, 6)) And IsNumeric(List1.Cells(i, 6)) And
        Not IsEmpty(List1.Cells(i, 7)) And IsNumeric(List1.Cells(i, 7)) Then

            Dim castka As String

            ws.Cells(radek, 2) = Format(List1.Cells(i, 3), "dd/mm/yyyy")
            ws.Cells(radek, 3) = List1.Cells(i, 2)
            ws.Cells(radek, 4) = List1.Cells(i, 6)
            ws.Cells(radek, 5) = List1.Cells(i, 7)
            castka = List1.Cells(i, 5)
            ws.Cells(radek, 6) = castka
            ws.Cells(radek, 7) = List1.Cells(i, 9)
        End If
    End If
End For
  
```

Obrázek 24 - Část kódu JUMP #3. Zdroj: Vlastní zpracování

Na obrázku 24 je vyobrazen kód, který řeší zmiňovanou situaci s výskytem jak účtu Má dáti tak i účtu Dal. V prvním řádku je spuštěn cyklus, který prochází vstupní data. Řídicí proměnnou je zde proměnná *i* typu integer, která je definována

2. Automatizace postupů při zpracování dat

v inicializátoru číslem sedm, které udává první řádek vstupních dat, na kterém je zahájeno zpracování dat. Na prvních šesti řádcích se nachází data, která nejsou relevantní. Jsou vyobrazena v tabulce 10, která byla graficky upravena, aby se vešla na stránku.

Účetní deník pro ZO 1,2,3,4,5,6,7,8 /2013 - Celý deník									

Řádek	Č.-Doklad	Datum	ZO	Částka	Účet MD	Účet Dal	Zak	Text	Vyhotoveno

Tabulka 10 - Ukázka prvních šesti řádku výstupu z účetního deníku systému JUMP. Zdroj: Výstupní data ze systému JUMP

Inkrementem for cyklu je next i na konci cyklu. Podmínkou pro skončení cyklu je dosažení řídicí proměnné počtu řádků na listu se vstupními daty. Ve vstupních datech se několikrát opakuje zobrazení popisků, která jsou zobrazena v tabulce 10. Tyto popisky jsou užitečné při analýze člověkem, ovšem počítač je musí umět přeskočit. Proto je v kódu uvedena podmínka, která dohlíží na to, aby v prvním sloupci byla hodnota číselná a aby buňka nebyla prázdná. MS Excel dokáže označit i prázdnou buňku jako číselnou a pro VBA se tato buňka tváří jako číslo. Skript musí hlídat, zdali se skutečně jedná o číslo a pokud ano, může pokračovat v analýze dat.

Aby mohla být splněna buňka diagramu z obrázku 23 s názvem Obsahuje řádek MD i D, je nutné ověřit, zdali jsou hodnoty v buňkách ve sloupcích Účet MD a Účet Dal číselné a neprázdné. Pokud tomu tak je, může dojít k zápisu dat do řádku nového listu.

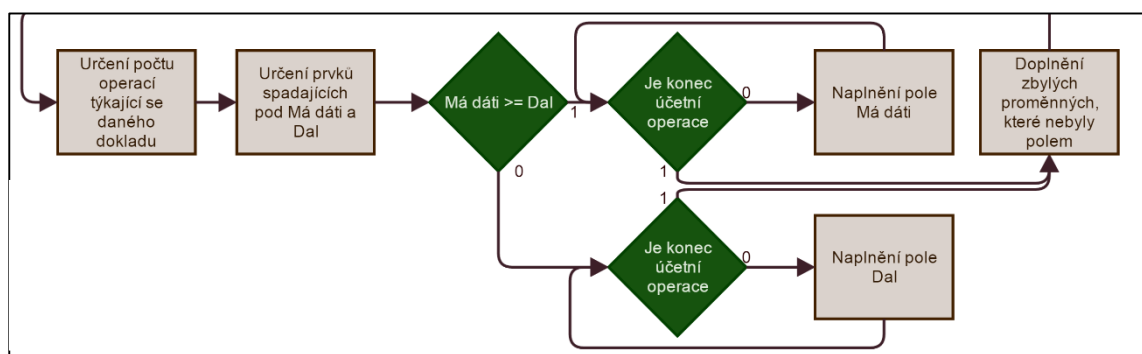
Formátováním data a následným zápisem dojde k zjednodušení v druhém sloupci nového listu. Do následujících buněk v daném řádku nového listu jsou pouze překopírovány data. U částky je potřeba jí nejprve uložit do proměnné castka, která je deklarována jako řetězec. Je to z důvodu nekompatibility s datovým typem double a integer. Integer nemůže být použit, protože se jedná o hodnotu s desetinným rozvojem. Double nemůže být použit, protože jsou hodnoty za desetinnou čárkou odlišovány čárkou nikoliv tečkou, kterou VBA k označení desetinného rozvoje využívá.

V případě složitějšího než pouze jednořádkového účtování se jednotlivé operace vyskytují ve více řádcích. V tabulce 9 je na prvních třech řádcích zobrazena operace, které bude věnován následující návrh a rozbor řešení.

Pokud není ve sledovaném řádku zdrojových dat současně hodnota ve sloupci Účet MD a hodnota ve sloupci Účet Dal, pak přechází skript k druhé větvi zpracování.

2. Automatizace postupů při zpracování dat

Zpracování takového zápisu je náročnější než samotné překopírování jako v minulém případě. Prvním krokem takového zpracování je určení počtu operací týkající se sledovaného čísla dokladu. Pokud je zjištěno, že sledovaný doklad má k sobě určitý počet operací, je zároveň dopočteno, že záznamů těchto operací bude na novém listu o jeden méně. To je způsobeno tím, že při vztažení určitého počtu hodnot většího než jedna jedné strany k pouze jedné hodnotě strany druhé dochází k přesunu samostatné hodnoty k hodnotám početnější strany a tím samostatná hodnota zaniká.



Obrázek 25 - Vývojový diagram popisující složitější z větví procesu zpracování dat ze systému JUMP. Zdroj: Vlastní zpracování

```
'
Zjisti kolik učetních operací se týká daného dokladu a zapis do nového listu
While (List1.Cells(i, 2) = List1.Cells(i + pocetUctovani, 2))
    pocetUctovani = pocetUctovani + 1
    ws.Cells(radek + pocetUctovani - 1, 3) = List1.Cells(i + pocetUctovani - 1, 2)
Wend
```

Obrázek 26 - Část kódu JUMP #4. Zdroj: Vlastní zpracování

Prvním procesem je cyklus, který určuje počet operací týkajících se daného dokladu. Tento cyklus je cyklem s podmínkou na začátku. Podmínka, která musí být splněna, se ptá, zdali je sledovaná buňka s číslem dokladu stejná jako první buňka s číslem dokladu. V každém opakování se sledovaná buňka posouvá o jednu níž a také se do proměnné pocetUctovani ukládá počet opakování. Tento cyklus by se dal řešit funkcí, která by byla volána vždy při splnění podmínky, kdy není v řádku vstupních dat k nalezení Účet MD a Účet Dal. Tento zápis je ovšem z důvodu lokálních a globálních proměnných a z důvodu možností modifikace kódu na jiné informační systémy v tomto případě vhodnější. Poslední operací, kterou tento cyklus vykonává, je zápis čísla dokladu do příslušné buňky.

Zpracování je navrženo tak, že rozhoduje počet výskytů syntetických účtů o tom, zdali je účtování proti jedné nebo druhé straně.

```

'      Pocitani pocetu prvku MD a D a ukladani do promennych
      For o = 1 To pocetUctovani

          If IsNumeric(List1.Cells(i + o - 1, 6)) And Not IsEmpty(List1.Cells(i + o - 1, 6))
Then
              pocetMD = pocetMD + 1
          End If

          If IsNumeric(List1.Cells(i + o - 1, 7)) And Not IsEmpty(List1.Cells(i + o - 1, 7))
Then
              pocetD = pocetD + 1
          End If

      Next o

```

Obrázek 27 - Část kódu JUMP #5. Zdroj: Vlastní zpracování

Zjištění, kolik jednotlivých účtů spadá pod kterou účetní stranu, je prováděno cyklem, který běží do počtu již zjištěné počtu operací týkajících se daného dokladu. Do proměnných pocetMD a pocetD je ukládán celkový počet výskytů. Tyto proměnné jsou deklarovány o úroveň výš, aby s nimi mohl pracovat i další cyklus či podmínka, které jsou dále ve skriptu uvedeny.

Zjištění počtu jednotlivých výskytů účtů je klíčové při rozhodování v dalším kroku, kterým je porovnání daných velikostí. Pokud je velikost proměnné pocetMD větší než velikost proměnné pocetD, je vytvořeno jednorozměrné pole řetězců MD (s velikostí pocetMD) a vytvořena proměnná typu řetězec DJeden.

```

'      Porovnání velikostí
      If pocetMD >= pocetD Then

          ReDim MD(pocetMD) As String
          ReDim D(pocetD) As String
          ReDim castkaMD(pocetMD) As String
          ReDim castkaD(pocetD) As String
          ReDim textMD(pocetMD) As String
          ReDim textD(pocetD) As String
          ReDim datumMD(pocetMD) As String
          ReDim datumD(pocetD) As String
          Dim MDJeden As String
          Dim DJeden As String

```

Obrázek 28 - Část kódu JUMP #6. Zdroj: Vlastní zpracování

U polí, která jsou vytvářena s velikostí udávanou prostřednictvím proměnné, se nepoužívá příkaz Dim, používá se ReDim. V obrázku 28 je vyobrazena tvorba více proměnných, které jsou přepisovány, pokud by podmínka byla splněná opačně.

V případě, že se objevilo ve vstupních datech více účtů na straně Dal než na straně Má dáti, byly by vytvořeny jiné proměnné.

Tyto proměnné jsou plněny cyklem, který opět projde všechny operace, které se týkají sledovaného čísla dokladu a naplní je hodnotami. Práce s polem je v jazyce Visual Basic stejná, jako je tomu v jazycích Java nebo PHP. Každý prvek pole je označen indexem. Tyto indexy nemusí nutně začínat nulou. Jednorozměrné pole můžeme deklarovat různým způsobem.

```
Dim Prodeje (2002 To 2009) As Double  
Dim Prodeje (1 To 8) As Double  
Dim Prodeje (7) As Double  
Dim Prodeje() As Double  
Dim Prodeje
```

Obrázek 29 - Část kódu VBA #1. Zdroj: [KRÁL, Martin. Excel VBA: výukový kurz. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.]

„Na prvním příkladu (viz obrázek 29) vidíte, že spodní hranice nemusí nutně začínat indexem 0 nebo 1. Pokud do pole ukládáte například roční hodnoty, může být výhodné pro čitelnost použít index korespondující s rokem.

Ve třetím příkladu bude mít pole osm prvků a je tato deklarace identická s deklarací (viz obrázek 30).“²⁷

```
Dim Prodeje (0 To 7)
```

Obrázek 30 - Část kódu VBA #2. Zdroj:[KRÁL, Martin. Excel VBA: výukový kurz. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.]

Pro účely skriptu bylo využito základního indexování začínající nulou. Takové indexování prvků pole bylo využito, protože velikosti polí jsou závislé na počtu operací připadající sledovanému číslu dokumentu a protože se může využít řídicí proměnná cyklu začínající nulou pro naplnění těchto polí.

Variantou využitou při plnění polí ve skriptu zpracovávajícím data ze systému JUMP bylo použití další celočíselné proměnné index, která se navyšuje při každém průchodu cyklem za určité podmínky. Podmínka je splněna, pokud je v buňce jedné strany hodnota číselná a pokud se v buňce hodnota vyskytuje. Při splnění těchto podmínek jsou do polí postupně ukládány jednotlivé hodnoty.

²⁷ KRÁL, Martin. Excel VBA: výukový kurz. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.

2. Automatizace postupů při zpracování dat

Do pole MD jsou ukládány hodnoty ze sloupce s jednotlivými názvy účtů. Do druhého pole castkaMD jsou ukládány příslušné částky odpovídající danému řádku. Do pole textMD jsou ukládány poznámky k jednotlivým operacím a do proměnné datumMD jsou ukládány jednotlivá data provedení operace.

```
Dim index As Integer
index = 0
For o = 0 To pocetUctovani - 1

    If IsNumeric(List1.Cells(i + o, 6)) And Not IsEmpty(List1.Cells(i + o, 6)) Then

        MD(index) = List1.Cells(i + o, 6)
        castkaMD(index) = List1.Cells(i + o, 5)
        textMD(index) = List1.Cells(i + o, 9)
        datumMD(index) = List1.Cells(i + o, 3)
        index = index + 1

    End If

    If IsNumeric(List1.Cells(i + o - 1, 7)) And Not IsEmpty(List1.Cells(i + o - 1,
7)) Then
        DJeden = List1.Cells(i + o - 1, 7)
    End If

Next o
```

Obrázek 31 - Část kódu JUMP #7. Zdroj: Vlastní zpracování

Pokud splněná podmínka o výskytu číselné hodnoty ve sledované buňce splněná není, jedná se o hodnotu z druhé účetní strany a tato hodnota je zapsána do proměnné DJeden.

Po skončení čísti skriptu na obrázku 31 jsou všechna pole naplněna daty. Posledním krokem je zápis těchto dat do nového listu (obrázek 32). Po zápisu do nového listu jsou pole v následujícím průchodu cyklem znovu deklarovány a jsou do nich uložena nová data, která se týkají následujícího dokladu a následující účetní operace.

```
'
Zápis do nového listu
For o = 0 To pocetUctovani - 2
    ws.Cells(radek + o, 4) = MD(o)
    ws.Cells(radek + o, 5) = DJeden
    ws.Cells(radek + o, 6) = castkaMD(o)
    ws.Cells(radek + o, 7) = textMD(o)
    ws.Cells(radek + o, 2) = Format(datumMD(o), "dd/mm/yyyy")
Next o
```

Obrázek 32 - Část kódu JUMP #8. Zdroj: Vlastní zpracování

2.3.2.3. Shrnutí

Zpracování dat prostřednictvím skriptu napsaném v jazyce VBA v prostředí aplikace MS Excel bylo v tomto případě velmi přínosné. Jednotlivé účetní případy byly v původních vstupních datech zapsány do více než jednoho řádku. To komplikovalo čtení a možnosti filtrování. Zpracováním těchto dat do vhodnější podoby bylo zjednodušeno čtení a usnadněno filtrování.

Zvolený postup byl navržen tak, aby procházel vstupní data několikrát a přepočítával jednotlivé výskyty účtů na obou účetních stranách. Volbou takového postupu se vyřešila problematika víceřádkového zápisu. Výsledný kód, který prochází data, je možné upravit na různé jiné výstupy z informačních systémů.

Byly využity pole řetězců, do kterých se ukládaly hodnoty z jednotlivých sloupců vstupních dat. Využití polí by mohlo být nahrazeno objekty. Objekt by obsahoval veškerá vhodná data ze vstupních dat a usnadnilo by to zpracování. Práce s poli byla v tomto případě využita, protože jejich využití umožňuje přeprogramovat skript na jinou strukturu dat. Využitím objektů by se určité části kódu zpřehlednily, ale jejich využití u jiné struktury by mohlo být nevhodné.

Skript v této podobě počítá s tím, že jsou data ve struktuře, která byla rozebrána na začátku oddílu a která je výstupem systému JUMP. Pokud by byl výstup ze systému JUMP pozměněn, skript by musel být přepsán, aby odpovídal aktuální struktuře.

2.3.2.4. Výstup

Po zpracování výstupních dat z informačního systému JUMP skriptem jsou data čitelnější a srozumitelnější. Nejdůležitějším faktorem je, že data po zpracování obsahují účty Má Dáti a Dal vždy na jednom řádku. Tento zápis umožňuje snadnější filtrování.

Data je v této podobě možné filtrovat podle názvů sloupců nebo podle hodnot v polích jednotlivých sloupců. Vyhledávání dat je přehlednější. Zajímavým způsobem zpřehlednění výstupních dat ze skriptu je obarvení jednotlivých účetních operací. Tato funkce je volitelná. Aby bylo dále usnadněno čtení, jsou veškeré hodnoty zapsány do jednotlivých buněk tak, aby řádky nebyly zalomovány.

Na následujícím obrázku 33 je vyobrazena podoba výstupních dat po zpracování skriptem.

Číslo řádku	Datum operace	Číslo dokladu	Má Dáti	Dal	Částka	Poznámka
1	03. 01. 2013	11-000002	343121	211000	79,68	brzd.váleček,řemen-3873
2	03. 01. 2013	11-000002	501506	211000	379,32	brzd.váleček,řemen-3873
3	03. 01. 2013	21-000001	325400	221100	1910,40	CCS CESKA SPOLECNOST
4	03. 01. 2013	21-000001	321000	221100	284010,00	FD 1- 861 Doprava a mec
5	03. 01. 2013	21-000001	321000	221100	7979,00	FD 1- 910 Beton Union P
6	03. 01. 2013	21-000001	321000	221100	24204,00	FD 1- 895 KOMERSIA WEST
7	03. 01. 2013	21-000001	321000	221100	11040,00	FD 1- 860 Doprava a mec
8	03. 01. 2013	21-000001	321000	221100	21000,00	FD 1- 808 MPI universal
9	03. 01. 2013	21-000001	321000	221100	45000,00	FD 1- 817 ABS-ELEKTRO s
10	03. 01. 2013	21-000001	321000	221100	8000,00	FD 1- 883 Květoň Pavel
11	04. 01. 2013	21-000002	221100	311100	11040,00	FV 1- 78 GOLDBECK Pref
12	07. 01. 2013	0-000002	527200	213000	6954,00	stravenky 183ks-1-2013
13	07. 01. 2013	0-000002	527200	213000	646,00	stravenky 17ks-1/2013
14	07. 01. 2013	0-000002	527200	213000	722,00	stravenky 19ks-1/2013
15	07. 01. 2013	0-000002	335100	213000	14048,00	stravenky 439ks-1/2013
16	07. 01. 2013	0-000002	527200	213000	8360,00	stravenky 220ks-1/2013
17	07. 01. 2013	11-000003	538000	211000	7500,00	dálniční známky 5ks
18	07. 01. 2013	11-000004	538000	211000	7500,00	dálniční známky 5ks
19	07. 01. 2013	11-000005	538000	211000	6000,00	dálniční známky 4ks
20	07. 01. 2013	21-000003	321000	221100	42805,00	FD 1- 931 STAVEBNINY LI
21	07. 01. 2013	21-000003	321000	221100	64411,00	FD 1- 929 STAVEBNINY LI
22	07. 01. 2013	21-000003	321000	221100	146011,00	FD 1- 930 STAVEBNINY LI
23	07. 01. 2013	21-000003	568100	221100	1285,88	bank.záruka-23820-Blovice
24	08. 01. 2013	21-000004	336100	221100	204822,00	OSSZ
25	08. 01. 2013	21-000004	336200	221100	53004,00	Všeobecná zdravotní
26	08. 01. 2013	21-000004	336200	221100	8387,00	Česká průmyslová zdr
27	08. 01. 2013	21-000004	336200	221100	13747,00	Oborová zdr.poj.
28	08. 01. 2013	21-000004	336200	221100	12651,00	Zdravotní pojišťovnaMV
29	08. 01. 2013	21-000004	342100	221100	54299,00	FÚ-daň z příjmu FO
30	08. 01. 2013	21-000004	325100	221100	20460,00	odpov.poj. z mezd 1Q/2013
31	08. 01. 2013	21-000004	379200	221100	2000,00	PP-KB penz.fond-VM
32	10. 01. 2013	11-000006	513000	211000	850,00	občerstvení-repré

Obrázek 33 - Výstup po zpracování dat ze systému JUMP skriptem. Zdroj: Vlastní zpracování

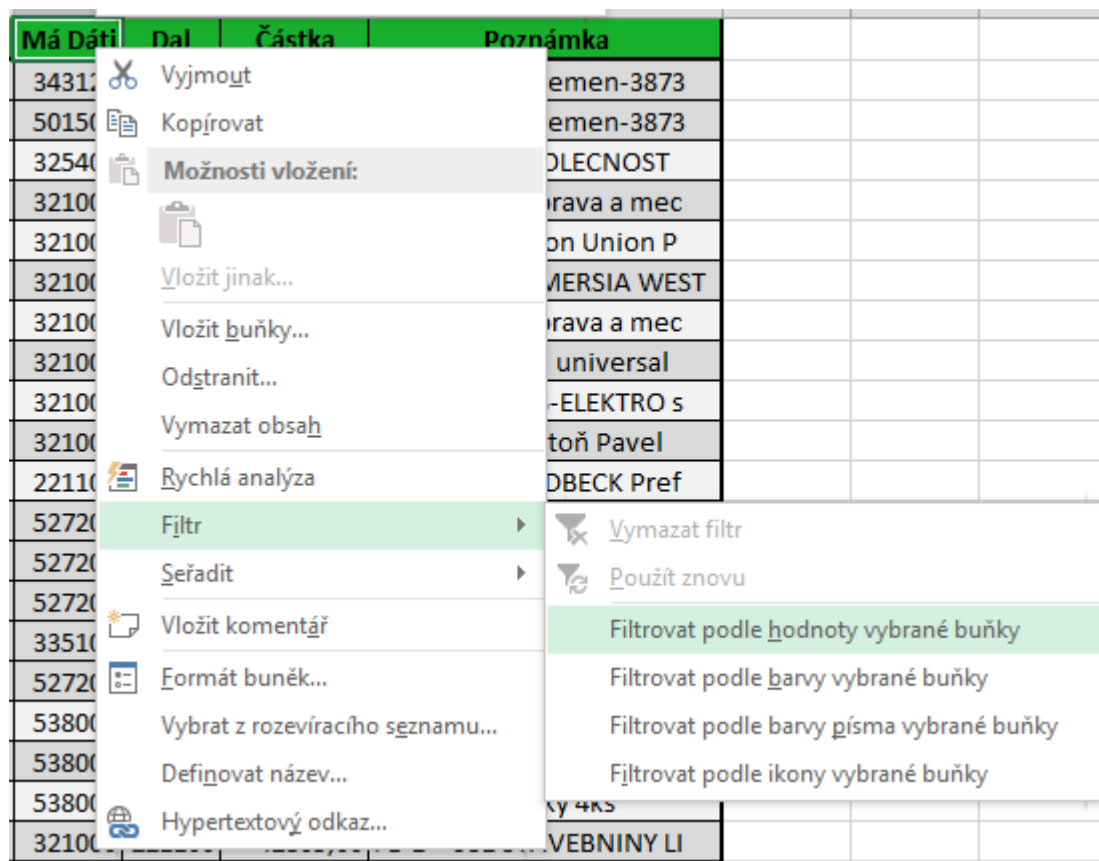
2.3.3. Postup zpracování

V části práce s názvem Způsob zpracování dat auditory bylo vysvětleno, jakým způsobem putují data mezi auditory a klienty auditorské společnosti. V okamžik, kdy přijdou data do ruky auditora, by měl mít auditor možnost je buďto zpracovat sám nebo by je měl mít již zpracované od někoho, kdo nad nimi spustí vytvořený skript. Pokud by data dostal v nezpracované podobě, musel by data otevřít v Excelu, otevřít vývojové prostředí Visual Basicu a spustit skript, kde by pouze zvolil název nového listu a kde by určil, o jaká data z jakého systému se jedná. Po provedení celého zpracování by se tato data dala snadno číst a filtrovat. Takové řešení je rozhodně efektivní. Jediným problémem by mohla být změna ve vstupních datech. Ta se ovšem v této práci nepředpokládá. Pokud

by data přišla již zpracovaná od někoho, kdo má zpracování na starost, nemusel by auditor tyto data dále zpracovávat. Možnost využít vytvořený skript má i zaměstnanec, který má za úkol tyto data pro auditory zpracovat. V případě, že zpracováním se zabývá někdo jiný než samotný auditor, je skript navržen tak, aby nebyla potřeba žádná nebo minimální znalost účetnictví a účetních operací. Pro spuštění skriptu potřebuje uživatel pouze novější MS Excel.

2.3.4. Výsledné filtrování

Po provedení skriptu, který automatizuje zpracování dat, je možné tyto data dále filtrovat, aby byla zpřehledněna oblast analýzy. Auditři mají často sadu svých vlastních pravidel, podle kterých v účetnictví vyhledávají vybrané účetní operace. K usnadnění vyhledávání mohou sloužit filtry v Excelu.²⁸



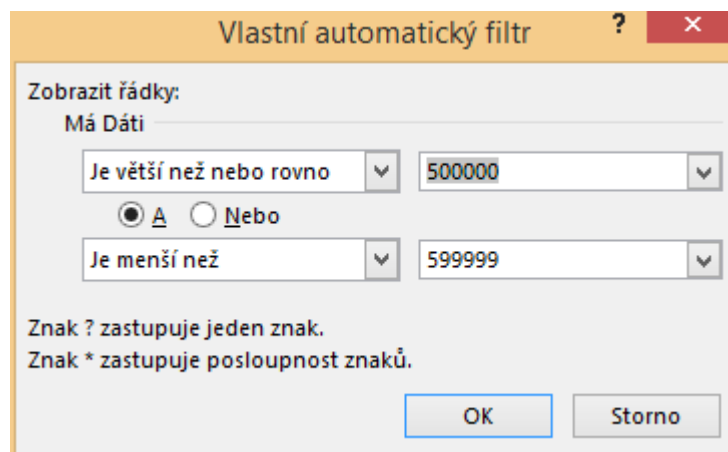
Obrázek 34 - Výběr filtru v MS Excelu. Zdroj: Vlastní zpracování, MS Excel 2013

Na obrázku 34 je zobrazeno jakým způsobem se vybírá filtr na sloupec. Stačí označit první buňku ve sloupci, stisknout pravé tlačítko, vybrat možnost filtr a zvolit jednu z možností filtrování. Možností filtrování je několik. Filtrování podle hodnoty

²⁸ LAURENČÍK, Marek. Excel 2010: práce s databázemi a kontingenčními tabulkami. 1. vyd. Praha: Grada, 2011, 168 s. Snadno a rychle (Grada). ISBN 978-80-247-3986-1.

2. Automatizace postupů při zpracování dat

vybrané buňky je v tomto případě nejvhodnější. Na sloupcích s hodnotami Má dáti a Dal je možné provést filtrování na základě vlastního filtru, kterým může být například vyhledání pouze nákladových účtů. Nákladové účty se pohybují v rozmezí mezi 500000 a 599999. První tři čísla označují syntetický účet a následující tři označují analytický účet. Na obrázku 35 je zobrazen vlastní filtr pouze nákladových účtů.



Obrázek 35 - Filtrování nákladových účtů. Zdroj: Vlastní zpracování, MS Excel 2013

Po potvrzení filtru z obrázku 35 zůstanou na listě pouze takové účetní operace, které se nacházejí na nákladových účtech na straně Má dáti. Přidáním filtru na účetní straně Dal je možné data dále filtrovat. Výsledný stav po filtru strany Dal, který byl nastaven na rovnost s hodnotou 211000, je zobrazen na obrázku 36.

	A	B	C	D	E	F	G
1	Číslo řád	Datum opera	Číslo dokla	Má D	Da	Částka	Poznámka
3	2	03. 01. 2013	11-000002	501506	211000	379,32	brzd.váleček,řemen-3873
18	17	07. 01. 2013	11-000003	538000	211000	7500,00	dálniční známky 5ks
19	18	07. 01. 2013	11-000004	538000	211000	7500,00	dálniční známky 5ks
20	19	07. 01. 2013	11-000005	538000	211000	6000,00	dálniční známky 4ks
33	32	10. 01. 2013	11-000006	513000	211000	850,00	občerstvení-repré

Obrázek 36 - Výsledek po filtrování. Zdroj: Vlastní zpracování, MS Excel 2013

Tímto způsobem je možné filtrovat různé položky a snadno vyhledávat v datech. Sloupec s datem operace je vhodný pro filtrování. Stejně tak může být vhodný sloupec s částkou, u které je možné vyhledávat například hodnoty větší než několik stovek korun. Bez předešlého zpracování do takové struktury by tyto filtry nebylo možné provádět.

3. Zhodnocení efektivnosti navrženého řešení

Z analýzy, která měla za úkol určit, která z možností řešení automatice je pro daný úkol nejvhodnější, jednoznačně vyšlo nejlepší řešení problému v jazyce Visual Basic for Application. Tento jazyk umožňuje data zpracovávat přímo v Excelu, aniž by byla potřeba překladač programu nebo pracování se souborem, jak to dělá jazyk Java. Z tohoto pohledu je navržené řešení nejefektivnější.

Skript je rozdělen podle jednotlivých vstupních dat. To znamená, že jednotlivá vstupní data mají každá svůj vlastní algoritmus a svůj vlastní kód, který se o zpracování stará. Vybraná vstupní data byla nejprve rozebrána, aby byla srozumitelná jejich struktura a aby bylo možné tyto data zpracovat. Byl navržen vývojový diagram, který vyobrazoval jednotlivé kroky zpracování dat. U výstupních souborů ze systému SAP byla data uložena do jednotlivých bloků, které nesly záznamy o účetních operacích. Zpracování těchto dat bylo nezbytné. Postup zpracování výstupních dat ze systému SAP může být rozšířen nebo přepsán pro potřeby jiných systémů. Tento postup se jeví jako efektivní v případě, kdy nebudou vstupní data nijak pozměněna. Skript byl napsán na taková data, jaká poskytla HZ Plzeň spol. s r. o. a jaká jsou přímo výstupem jisté nejmenované společnosti. Systém SAP nabízí určité možnosti konfigurace výstupů, na které by se skript dal přeprogramovat. U výstupních souborů ze systému JUMP byla data uložena do jednotlivých řádků, ale jisté účetní operace byly rozepsány do více než jednoho řádku. Zpracování těchto dat bylo též nezbytné. Postup zpracování výstupních dat ze systému JUMP může být rozšířen nebo přepsán pro potřeby jiných systémů stejně jako postup zpracování výstupních dat ze systému SAP. Tyto dva návrhy zpracování mohou být společně propojeny. Každý řeší odlišný problém, ale výstupem jsou podobná data. Skript byl také napsán na taková data, jaká poskytla HZ Plzeň spol. s r. o. a jaká jsou přímo výstupem jisté nejmenované společnosti.

Následnou filtrací je dosaženo požadované podoby dat. Pomocí filtrů je možné data procházet efektivně. Tyto filtry se dají konfigurovat libovolně podle potřeb uživatele. Kombinace automatizace pomocí skriptu a filtru je efektivní pro usnadnění analýzy zpracovaných dat.

Závěr

Hlavním cílem práce bylo navrhnout řešení automatizace postupů při zpracování dat z účetních deníků pro účely účetního auditu.

V teoretické části byly po uvedení do problematiky představeny základní pojmy z oblasti informačních systémů, informatiky a programování. Byly charakterizovány informační systémy, účetní softwary a programovací jazyky. Následně byla charakterizována problematika, ve které byl podrobněji rozebrán problém, kterým se práce zabývá. Další část byla věnována problematice roztríštěnosti informačních systémů a rozdělení trhu s informačními systémy, podle velikostí organizací, které tyto informační systémy využívají. Poslední díl teoretické části byl věnován vysvětlení, jakým způsobem v auditorské společnosti pracují s daty a jakým způsobem tyto data podnikem putují.

V praktické části byla rozebrána výstupní data z vybraných účetních softwarů nebo informačních systémů. Byly charakterizovány tři vybrané struktury dat. Dále byla představena vhodná struktura dat, která je vhodnější než původní výstupní data ze systémů. Následující část se zabývala možnostmi řešení automatizace. Byly navrženy tři možnosti řešení automatizace, kterými bylo vytvoření skriptu v programovacím jazyce Visual Basic for Application v prostředí aplikace MS Excel, naprogramování programu v programovacím jazyce Java a vytvoření či naprogramování jednotlivých rozšíření výstupních modulů v informačních systémech. Z těchto možností byl vybrán VBA jako vhodný nástroj pro zpracování dat.

Hlavní částí celé práce a praktické části byl samotný návrh a dokumentace skriptu, který dokáže vhodně data zpracovat. Byla rozebrána vstupní data ze systémů SAP a JUMP a byla navržena řešení zpracování, která byla rozkreslena do vývojových diagramů. Dále byl celý skript zdokumentován a rozebrány úryvky z kódu. Po dokončení dokumentace návrhu skriptu byl návrh řešení shrnut a byl zobrazen výstup. Následně byly zobrazeny možnosti filtrování zpracovaných dat. V poslední části práce byla zhodnocena efektivita navrženého řešení a byly vysvětleny některé důvody a postupy z praktické části.

Seznam obrázků

Obrázek 1 - Diagram popisující způsob zpracování dat auditory	19
Obrázek 2 - Prostředí pro zpracování jazyka Java.	26
Obrázek 3 - Vestavěné primitivní datové typy jazyka Java.	27
Obrázek 4 - Vý. diag. popisující celý proces zpracování dat ze systému SAP.	31
Obrázek 5 - Vý. diag. popisující úvodní část zpracování dat ze systému SAP.	32
Obrázek 6 - Ukázka grafického uživatelského prostředí pro ovládání skriptu.	32
Obrázek 7 - Část kódu SAP #1.	33
Obrázek 8 - Vý. diag. zobrazující pojmenování sloupců při zpracování dat (SAP).	33
Obrázek 9 - Část kódu SAP #2.	34
Obrázek 10 - Vý. diag. popisující hlavní proces zpracování dat ze systému SAP.	34
Obrázek 11 - Část kódu SAP #3.	36
Obrázek 12 - Část kódu SAP #4.	37
Obrázek 13 - Část kódu SAP #5.	37
Obrázek 14 - Část kódu SAP #6.	37
Obrázek 15 - Část kódu SAP #7.	38
Obrázek 16 - Stav B	39
Obrázek 17 - Stav A	39
Obrázek 18 - Výstup po zpracování dat ze systému SAP skriptem.	40
Obrázek 19 - Vý. diag. popisující celý proces zpracování dat ze systému JUMP.	42
Obrázek 20 - Vý. diag. popisující úvodní procesy zpracování dat ze systému JUMP.	43
Obrázek 21 - Část kódu JUMP #1.	43
Obrázek 22 - Část kódu JUMP #2.	43
Obrázek 23 - Vý. diag. popisující jednu z větví procesu zpracování dat (JUMP).	44
Obrázek 24 - Část kódu JUMP #3.	44
Obrázek 26 - Vý. diag. popisující složitější z větví procesu zpracování dat (JUMP).	46
Obrázek 25 - Část kódu JUMP #4.	46
Obrázek 27 - Část kódu JUMP #5.	47
Obrázek 28 - Část kódu JUMP #6.	47
Obrázek 29 - Část kódu VBA #1.	48
Obrázek 30 - Část kódu VBA #2.	48
Obrázek 31 - Část kódu JUMP #7.	49
Obrázek 32 - Část kódu JUMP #8.	49
Obrázek 33 - Výstup po zpracování dat ze systému JUMP skriptem.	51
Obrázek 34 - Výběr filtru v MS Excelu.	52
Obrázek 35 - Filtrování nákladových účtů.	53
Obrázek 36 - Výsledek po filtrování.	53

Seznam tabulek

Tabulka 1 - Podniky používající vybrané IS; leden 2014 (hodnoty v %).	16
Tabulka 2 - Ukázka účetního deníku jako výstupních dat ze systému SAP.	21
Tabulka 3 - Ukázka účetního deníku jako výstupních dat ze systému K2.	21
Tabulka 4 - Ukázka výstupních dat ze systému JUMP.	22
Tabulka 5 - Ukázka vhodné podoby výstupních dat po automatickém zpracování.	22
Tabulka 6 - Přehled typů proměnných.	24
Tabulka 7 - Výtažek z účetního deníku jako výstupních dat ze systému SAP.	30
Tabulka 8 - Hlavička výstupních dat ze systému JUMP.	41
Tabulka 9 - Vstupní data ve zjednodušené a relevantní podobě.	41
Tabulka 10 - Ukázka prvních šesti řádků výstupu z účetního deníku systému JUMP.	45

Seznam grafů

Graf 1 - Vývoj přírůstku českého ERP trhu v letech 2005 až 2011. Graf znázorňuje každoroční relativní přírůstek nových ERP projektů (referencí) ve všech segmentech trhu. _____	15
Graf 2 - Hodnoceno 62 all-in-one ERP systémů nasazených v malých organizacích v ČR (od 10 do 49 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 9 829 referencí. _____	17
Graf 3 - Hodnoceno 62 all-in-one ERP systémů nasazených ve středně velkých organizacích v ČR (od 50 do 249 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 7 324 referencí. _____	17
Graf 4 - Hodnoceno 62 all-in-one ERP systémů nasazených ve velkých organizacích v ČR (od 250 do 1 000 zaměstnanců) do konce roku 2011. Tento segment zahrnuje celkem 3 281 referencí. _____	18

Seznam použité literatury

Publikace:

- [1] ANDERSON, George W. *Naučte se SAP za 24 hodin*. 1. vyd. Brno: Computer Press, 2012, 432 s. ISBN 978-80-251-3685-0.
- [2] BASL, Josef a Roman BLAŽÍČEK. *Podnikové informační systémy: podnik v informační společnosti*. 3., aktualiz. a dopl. vyd. Praha: Grada, 2012, 323 s. Management v informační společnosti. ISBN 978-80-247-4307-3.
- [3] DRUCKER, Peter F. D. *Postkapitalistická společnost*. 1.vyd. Praha: Management Press, 1993, 197 s. ISBN 80-856-0331-4.
- [4] FUCHS, Joachim a Andreas BARCHFELD. *Visual Basic: velká kniha řešení*. Vyd. 1. Překlad Jiří Kolář, Milan Daněk. Brno: Computer Press, 2010, 722, xxii s. Programování (Computer Press). ISBN 978-80-251-2212-9.
- [5] KRÁL, Martin. *Excel VBA: výukový kurz*. Vyd. 1. Brno: Computer Press, 2010, 504 s. ISBN 978-80-251-2358-4.
- [6] LAURENČÍK, Marek. *Excel 2010: práce s databázemi a kontingenčními tabulkami*. 1. vyd. Praha: Grada, 2011, 168 s. Snadno a rychle (Grada). ISBN 978-80-247-3986-1.
- [7] LAURENČÍK, Marek a Michal BUREŠ. *Programování v Excelu 2010: záznam, úprava a programování maker*. 1. vyd. Praha: Grada, 2013, 198 s. Průvodce (Grada). ISBN 978-80-247-5033-0.
- [8] MACEK, Jan. *Ekonomická a sociální statistika*. 1. vyd. Plzeň: Západočeská univerzita v Plzni, 2008, 240 s. ISBN 978-80-7043-642-4.
- [9] MACEK, Jan, Rudolf KOPEK a Jitka SINGEROVÁ. *Ekonomická analýza podniku*. 1. vyd. V Plzni: Západočeská univerzita v Plzni, 2006, 157 s. ISBN 80-704-3446-5.
- [10] SCHILDT, Herbert. *Java 7: výukový kurz*. 1. vyd. Brno: Computer Press, 2012, 664 s. ISBN 978-80-251-3748-2.
- [11] SODOMKA, Petr a Hana KLČOVÁ. *Informační systémy v podnikové praxi*. 2. aktualiz. a rozš. vyd. Brno: Computer Press, 2010, 501 s. ISBN 978-80-251-2878-7.

[12] SPELL, Brett. *Java: programujeme profesionálně*. Vyd. 1. Překlad Bogdan Kiszka. Praha: Computer Press, 2002, 1022 s. Programujeme profesionálně. ISBN 80-722-6667-5.

[13] WEBER, Monika. *Excel VBA: velká kniha řešení*. 1. vyd. Brno: Computer Press, 2007, 867 s. ISBN 978-80-251-1453-7.

Elektronické zdroje:

[14] Čo je program Excel?. Support Office. [online]. [2015] [cit. 2015-03-25]. Dostupné z: <https://support.office.com/sk-sk/article/%C4%8Co-je-program-Excel-8373c3d7-bd64-4b7f-bdbd-1fa4b2007b09?ui=sk-SK&rs=sk-SK&ad=SK>

[15] SODOMKA, Petr a Hana KLČOVÁ. *Český trh ERP zrychlil růst: V malých a středních podnicích přibylo dva tisíce projektů*. [online]. [cit. 2015-02-24]. Dostupné z: <http://www.systemonline.cz/erp/cesky-trh-erp-zrychlil-rust.htm>

[16] Visual Basic. Microsoft Development Network. [online]. [2013] [cit. 2015-03-22]. Dostupné z: <https://msdn.microsoft.com/cs-cz/library/2x7h1hfk.aspx>

[17] What is Java technology and why do I need it?. Java.com. [online]. [2015] [cit. 2015-03-21]. Dostupné z: https://www.java.com/en/download/faq/whatis_java.xml

[18] WOOD, Bill. *ERP vs. ERP II vs. ERP III Future Enterprise Applications*. [online]. [cit. 2015-03-15]. Dostupné z: <http://www.r3now.com/erp-vs-erp-ii-vs-erp-iii-future-enterprise-applications/>

Abstrakt

STELZER, J. Automatizace postupů při zpracování dat z účetních deníků pro účely auditu. Bakalářská práce. Plzeň: Fakulta ekonomická ZČU v Plzni, 62 s., 2015

Klíčová slova: zpracování dat, automatizace, Visual Basic, programování, informační systémy, audit

Tématem bakalářské práce je automatizace postupů při zpracování dat z účetních deníků pro účely auditu. Hlavním cílem této práce je navržení automatizace zjednodušující výstupy z účetních deníků. K dosažení tohoto cíle bylo nutné definovat základní pojmy z oblasti informačních systémů a charakterizovat problematiku roztržitosti informačních systémů a ekonomických softwarů. Tyto definice jsou popsány v teoretické části práce. V praktické části práce jsou rozebrána vstupní data a jejich podoba. Jsou rozebrány vybrané možnosti automatizace zpracování dat a je navržen skript v jazyce Visual Basic, který data zpracovává. Efektivnost zvoleného řešení a možná rozšíření jsou popsány v poslední části práce. Pro lepší přehled jsou použity grafy, tabulky, obrázky, části kódu a vývojové diagramy. Na základě zjištěných se došlo k závěru, že popsáný návrh řešení je efektivní.

Abstract

STELZER, J. Bookkeeping journal's data processing automation for financial audit. Bachelor's thesis. Pilsen: Faculty of Economics, University of West Bohemia, 62 pages, 2015

Key words: data processing, automation, Visual Basic, programming, information system, audit

The topic of my bachelor thesis is named Bookkeeping journal's data processing automation for financial audit. The main goal of my thesis is design automation that will simplify output data from bookkeeping journals. To achieve this goal it was necessary define the basic concept of information systems and describe fragmentation issue of information systems and economic software. These definitions are described in the theoretical part. In practical part is analysis of the output data and their structure. In practical part are also analyzed selected options of data processing automation and designed script in Visual Basic, which processes these data. The efficiency of selected solution and possible extension are described in the last part. For better understanding there were added graphs, charts, pictures, parts of code and flowcharts. Based on previous information the selected solution can be described as a successful one.